(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) Publication number: **0 388 574 B1**

(12) # EUROPEAN PATENT SPECIFICATION

(45) Date of publication of patent specification: **15.06.94** (51) Int. Cl.5: **H04L 12/40**

(21) Application number: **89810229.8**

(22) Date of filing: **23.03.89**

(54) **Method and apparatus for distributed queue multiple access in a communication system.**

(43) Date of publication of application:
26.09.90 Bulletin 90/39

(45) Publication of the grant of the patent:
15.06.94 Bulletin 94/24

(84) Designated Contracting States:
DE FR GB

(56) References cited:
EP-A- 0 138 717
WO-A-86/03639
CA-A- 1 158 739

I.E.E.E. COMMUNICATIONS MAGAZINE, vol. 26, no. 4, April 1988, pages 20-28, IEEE, New York, NY, US; R.M. NEWMAN et al.: "The OPSX man"

(73) Proprietor: **International Business Machines Corporation**
Old Orchard Road
Armonk, N.Y. 10504(US)

(72) Inventor: **Nassehi, Mehdi M.**
Buenistrasse 16
CH-8134 Adliswil(CH)
Inventor: **Müller, Hans R.**
Erlenweg 7
CH-8135 Langnau am Albis(CH)
Inventor: **Zurfluh, Erwin A.**
Heerenstrasse 24
CH-8706 Feldmeilen(CH)
Inventor: **Bux, Werner**
Reidholzstrasse 33
CH-8805 Richterswil(CH)
Inventor: **Zafiropulo, Pitro A.**
Mühlestrasse 20
CH-8803 Rüschlikon(CH)

(74) Representative: **Barth, Carl Otto et al**
IBM Corporation
Zurich Intellectual Property Dept.
Säumerstrasse 4
CH-8803 Rüschlikon (CH)

EP 0 388 574 B1

Description

FIELD OF INVENTION

Present invention is concerned with access to the transmission medium in a communication network comprising two busses and a plurality of stations connected between them, and in particular to a method and apparatus for regulating access to the transmission medium on the basis of distributed request queues.

BACKGROUND

Several kinds of networks are known in which plural nodes or stations gain access to a common transmission medium. Examples of such systems are single bus networks with collision detection, and token ring networks as well as token passing bus networks. Recently, networks providing another technique of multiple access to a common transmission medium have become of interest. These are systems comprising two parallel busses with counterflowing transmission of information on the two busses. Slots are released at regular intervals by headend stations, and these slots are used by the node stations for data transmission. Each node station has to request access to a slot by previously transmitting an access request in an Access Control Field of a passing slot. It keeps a count of access requests it has seen from other stations (located upstream in request transmission direction) before it raises an own access request, and lets as many free slots pass by (for use by the other stations) as the count indicated, before it occupies the next free slot for transmission of its own data.

Such systems were described e.g. in a paper by R.M.Newman et al. entitled "The QPSX MAN", published in the IEEE Communications Magazine, Vol.26 No.4 (April 1988) pp.20-28; and in a Draft Proposed IEEE Standard 802.6 entitled "Distributed Queue Dual Bus (DQDB) Metropolitan Area Network (MAN)", Draft D6, November 15, 1988.

Though these known distributed queueing systems are well suited for networks comprising a limited number of stations, they have some disadvantages which become inacceptable and may render the system inefficient when the number of stations is raised to several hundred, and when the length of the transmission busses is in the order of several kilometers.

These disadvantages are in particular: An "unfairness" for some stations with respect to others, due to the fact that each station has to await a free access request field before it can transmit a request, so that stations located upstream (in request transmission direction) are preferred; and an impossibility to guarantee the availability of a sequence of consecutive slots for one station desiring to transmit the portions of a data packet without interruption.

OBJECTS OF THE INVENTION

It is an object of the invention to provide, in a system with two busses and distributed request queueing, an access technique which eliminates unfairness between stations.

It is another object of the invention to guarantee in a distributed request queueing communication system the availability of a sequence of consecutive free transmission slots for a station which previously requested at one time the respective number of slots.

SUMMARY OF THE INVENTION

These objects are achieved by the method and apparatus of the invention which provide, in a distributed queueing system, the ability to request multiple slots at a time, and the storage of external requests seen from other stations together with local requests, both in the form of request counts, in a single sequential request queue (FIFO type storage). They provide further an individual delay for request counts entering the request queue in a station, in such a way that the influence of different propagation times as seen for the different stations along the busses is eliminated (which would otherwise result in an inconsistency between the distributed request queues so that the stations would behave differently with respect to any particular slot propagating along a bus).

The invention ensures, even for a large number of stations and in a very long bus network, fairness among stations, i.e. giving each station equal chances for obtaining access to the transmission medium for data transfers. It further guarantees that when a station requested, by a single request, a plurality of slots, it will see an uninterrupted sequence of free slots as requested once it started transmitting, so that the necessity for numbering the different segments of one data packet and for providing housekeeping

procedures to reassemble them (because they may not arrive consecutively at their destination) is eliminated.

These and other advantages will become more apparent from the following description and illustrative drawings of a preferred embodiment of the invention.

LIST OF DRAWINGS

Fig.1A is a schematic representation of a dual bus system in which the invention can be used;
Fig.1B is a schematic representation of a folded bus system in which the invention can be used;
Figs. 2A - 2D show the representation of data on the busses, in particular a cycle frame (2A) including several slots, a slot (2B) with its various fields, and and two forms of an access control field of a slot (2C, 2D);
Fig.3 shows the request queue storage of a station according to the invention, plus related circuitry;
Fig.4 schematically shows the request queues of two stations, and illustrates the propagation time differences for the requests before they enter the request queues;
Fig.5 shows the request queue of a station in connection with a delay element according to the invention,
Fig.6 schematically shows the request queues and their associated delay elements, in two stations,
Fig.7 (combination of Figs.7A and 7B) is a block diagram of a station's request queue and bus access circuitry, embodying the invention;
Fig.8 is a block diagram of one bus controller of the circuitry of Fig.7;
Fig.9 is a block diagram of another bus controller of the circuitry of Fig.7;
Fig.10 shows the wrap-around modifications for a dual bus system which enable the initialization of individual delays in the stations by the transmission of special markers, and
Fig.11 is a block diagram of front-end circuitry interconnecting each station with the two busses, including signal conversion and clock signal derivation.

DETAILED DESCRIPTION

1) Environment System (Basic Network)

Fig.1A and 1B show two forms of a communication network in which present invention finds application. Both networks consist of two busses (or bus segments) with a number of stations (nodes) connected between the two busses. Information on each bus is flowing in one direction only, and on the two busses it is counterflowing (antiparallel).

The network of Fig.1A referred to as dual bus configuration comprises two separate busses A (11) and B (13). A headend station is connected to each bus, i.e. headend station HE-A (15) to bus A and headend station HE-B (17) to bus B. Each headend station releases time slots (explained below) for use by the node stations (19-1, 19-2, ... , 19-N) along the bus for data transmission. A special request technique, "distributed queueing", is used for regulating access by the stations to slots on the bus. The dual bus system and the access technique (which will be briefly reviewed in the sequel) were described in the paper and in the standards proposal already mentioned above in the introduction.

Each bus could end at the node station remote from the associated headend station, without being connected to the other headend station. However, some connection from the last node station (e.g. N) to the other headend station (e.g. HE-B) shown as line without arrow (21, 23) can be provided for special purposes.

The network shown in Fig.1B referred to as folded bus configuration consists essentially of plural node stations (25-1, 25-2, ... , 25-N), a single headend station (27) and a folded bus comprising two bus segments called bus A and bus B, or outbound bus (31) and inbound bus (33), respectively. A fold connection (29) is provided at the last station N for interconnecting the two bus segments. The headend station is connected to both bus segments for releasing time slots to the outbound bus (A) and receiving returning time slots on the inbound bus (B). (In principle, the connection of the inbound bus from the first station back to the headend station, designated as 35 in the drawing, could be omitted in a normal folded bus system but here it is assumed that such connection exists). As in the dual bus network of Fig.1A, information on the two bus segments is counterflowing. The same access mechanism (distributed queueing) can thus also be used on this folded bus system.

For the ease of description, the term "bus" is used in the following for each of the two separate busses of a dual bus system as well as for each of the two bus segments of a folded bus system.

In the following, the invention will be explained only in connection with a folded bus system as shown in Fig.1B, to avoid any complication in the description due to the fact that everything is duplicated on the dual bus system of Fig.1A (requests on bus A for access to bus B, and requests on bus B for access to bus A). It should be noted however that the invention is also well suited for such dual bus systems, and any modifications necessary will be explained at the end of the specification.

Frame/Slot Structure:

Figs. 2A ... 2D show the structure of the information on the busses. The basic unit is the slot. Each headend station generates such slots which are of fixed length, at regular intervals. For synchronization and control purposes, a cycle frame may be defined as shown in Fig.2A. The beginning of the cycle frame is a cycle frame header which contains information on synchronization, overall control, configuration, etc. The header is followed by consecutive slots.

The slot format can be seen in Fig.2B. Each slot has an Access Control Field (ACF) for control information, and a data segment field for carrying the data to be transmitted. Of course, the data segment contents may also be subdivided into a header portion and a data payload portion. As such subdivision is not relevant for present invention, the data segment will be considered as an entity in the following description. A typical size of a slot is e.g. 70 bytes; two bytes for the ACF and the remaining 68 bytes for the data segment field. This can of course be selected according to the requirements of the respective system.

The access control field is shown in more detail in Fig.2C. As mentioned above, in a typical example it comprises 2 bytes or 16 bits. The first field carries the Busy/Free (B/F) bit indicating whether the data segment actually contains data or not. A station can access a slot for data transmission only if this bit indicates that the slot is still empty. The next field carries a Type (TYP) bit which indicates to which of two different categories (e.g. arbitrated or non-arbitrated) the slot belongs. The next field (PRI) comprises two bits indicating the priority to which the slot belongs. It is assumed for the system to be described as embodiment of the invention that four priority categories are provided which can be distinguished by these two bits. A Reserved (RES) field comprises four bits which are not utilized in the main example to be described and can be assigned to any purpose by the system designer. (The possibility to transfer special markers in this field is mentioned in section 7 below). The field REQUEST comprises eight bits to be used for access requests. In the present case, this field can contain a request count for up to 256 slots. It is assumed here that per ACF, a request can be made only for one priority category, as determined by the two bits in the Priority field. The headend station sets the value in the Priority field cyclically to one of the four categories such that in the total sequence of slots on the busses, one quarter is associated with each one of the priorities.

Any other format of the ACF can of course be chosen. As one alternative, Fig.2D shows an ACF for a two-priority system. The first ACF bit again indicates the Busy/Free status, the second bit determines the slot Type. Two further bits are Reserved. The remaining 12 bits are subdivided into two request fields (REQ 0, REQ 1), each for one of the two priorities considered. Thus, each field can accomodate a six-bit request count so that for each priority a request for up to 64 slots can be made. For the following description, the ACF format of Fig.2C is assumed.

2) Essentials of Prior Art and Invention

Before explaining the invention, the state of the art of dual bus systems with distributed queueing access will be briefly reviewed. Such known systems are described e.g. in the publications already mentioned in the introduction.

In the known systems, the ACF comprises four one-bit request fields, each for one priority. Thus, a request for a single slot can be made per priority in each ACF. A station requiring access to a slot (in a given priority class) waits for a free request field in that priority, and inverts the respective bit to indicate its access request to the other stations. It keeps a count of access requests it has seen from other stations before making its own request, and lets as many free slots pass for the other stations as it had seen external access requests prior to its own, so that they can be correctly served. Only then, it accesses the next free slot passing its access point. In the meantime, it has again counted the external access requests seen from other stations, and keeps on accumulating these external access requests until it places its own next request.

This procedure has several disadvantages.

1) Only one slot can be requested at a time.

2) The next access request can only be made if the previous one has been served (i.e. there must not be more than one pending access request in any station).

3) A local request is registered in the station's request queue (actually by transferring the external request count from one counter to another) as soon as the local request is generated in that station. However, there may elapse a considerable time interval before that station can enter its local request into a passing slot (if a free request subfield does not appear for some time because of high activity of other stations). This results in an unconsistency between the queues of different stations.

4) Due to different propagation times (delay) of the ACF fields with the requests in one direction and of the free slots for data transmission in the other direction, there may be discrepancies between the time when a local request comes up to the top of the waiting queue (i.e. when all external request which were older than the local request have been served in the opinion of the respective station), and when the "corresponding" free slot arrives at the station's access point. In other words, in certain situations a station P may grab away a free slot which actually was reserved for a station Q, because the station P believed that it was now its turn to access a slot for data transmission.

The facts of points 1, 2, and 3 cause "unfairness" in the system, i.e. the node stations have unequal chances or different probabilities to access a free slot for data transmission.

The facts mentioned in point 4 prevent the possibility to guarantee to any station the availability of a sequence of consecutive slots which is often desirable to enable continuous transmission of a larger data packet. Even if multiple slot reservations were allowed at one time, the availability of a string of consecutive free slots could not be guaranteed to a station having made the correct reservations, because of the facts mentioned above. As a result, the system must provide for the segmentation of larger data packets into slot-size segments that can be transmitted independently; this necessarily results in increased overhead, i.e. additional administrative steps and data transmissions for segmenting and reassembling the data packets.

The invention avoids these disadvantages by the following features:

a) The reservation of multiple slots at a time is enabled by providing count fields for reservation.

b) Further requests can be made by a station even if a pending request was not yet served. Thus, a plurality of pending requests each for multiple slots are enabled. (However, an upper limit or request window size may still be set by the system.)

c) A local request is only entered into a station's request queue after this request has been actually transmitted in the request subfield of a passing slot.

d) Both, the external requests seen by a station and the own local requests, are stored in a single sequential queue clearly indicating the sequence of the requests (and their temporal relationship).

e) A selective delay is introduced to eliminate the influence of the different propagation times for requests and free slots on the two busses as seen for different stations.

These measures guarantee that all stations independent of their location along the busses are treated equally fair, and that the access to a consecutive sequence of requested slots can be guaranteed for each station, thus eliminating the complexity of data packet segmentation and reassembly.

The various features of the invention are explained generally in the following section. Later, an actual implementation will be described.

3) Details of Inventive Features

Some essential features of present invention will now be explained with reference to Fig.2C and Figs.3-6.

In Fig.2C (and 2D) it has already been shown that request fields are provided which can each accomodate a request count, i.e. a number representing a multiple request for a plurality of (consecutive) slots.

Fig.3 is a block diagram of a request queue in the form of a FIFO (First-In First-Out) storage, and its related elements in a station. This block diagram contains only the features necessary for describing some of the basic inventive functions; a more complete description is given later in connection with Fig.7.

The request queue FIFO storage is schematically shown at 41. It comprises sequential storage positions 43-1 to 43-N each for holding either an external request count EXT-REQ or a local request count LOC-REQ, both representing a number as can be held in the request field of a slot's ACF, plus an indication whether the request is external or local. The contents of the top position (43-1) can be decreased by one unit when a passing empty slot is detected on bus A by detecting means 45 (TEST B/F).

The contents of each request field (RQ-F) of passing slots are extracted by selection means 47 (SEL) and inserted into the bottom position 43-N of the FIFO storage as request count. If it was detected by detecting means 49 (TEST RQ-F) that the external request count arriving in a slot on bus B is zero, then a

local request waiting in a buffer 51 is gated through gate 53 into the empty request subfield of the passing slot on bus B as local request count. Simultaneously a LOC-REQ indication is sent on line 55 to the FIFO request queue entry to mark the value just being inserted into the FIFO request queue as local request count. Thus, the request queue in FIFO 41 contains all the access request counts as seen by the respective station or generated by itself, in the correct sequential order.

Whenever an empty slot is detected on bus A, the existing count in the top position is decremented by one. If the count is for an external request, the free slot is not accessed but left free for another station downstream. When the top position has reached count zero, then the next position becomes the top position. If there is a count in the top position which represents a local request, as will be indicated on line 57 (LOC-REQ), then a signal is furnished through AND gate 59 as "set busy" signal to the bus and as release signal to gating means 61 which will then gate the data waiting in buffer 63 into the data segment field of the passing slot on bus A. This will continue until the local request count in the top position has reached the value zero. Of course, instead of having a separate FIFO storage, the request queue can be implemented in a RAM store with pointers. The various gating and control means can also be implemented in different ways.

It should be noted that in this configuration, access requests are inserted on and copied from bus B, whereas data are inserted on bus A but copied from bus B.

Fig.4 illustrates the situation for the queues in two selected stations i and j. Though the queues have different contents, they correctly reflect the access situation for each individual station. As is also already indicated in Fig.4, the propagation delay for any slot from the bus A access point of station i to its entry point into the FIFO queue is different from the propagation delay for station j ($Di \neq Dj$). This will result in the fact that when any single slot request which was inserted into both queues reaches the top position of the queue e.g. in station i, this station may see passing on bus A another slot than station j sees when the corresponding single slot request reaches the request queue top position in that station.

The elimination of this difference is one feature of the invention and will now be explained with reference to Fig.5. In each station, in addition to the FIFO request queue 41, there is provided a delay element 71, by which a selective delay which is individual to the respective station can be introduced. The delay is so selected (as will be explained in the sequel) that the propagation time of a slot from the bus A access point (67) to the entry (69) of the FIFO request queue is equal for all stations of the system. Principles of the operation of the delay means are illustrated in Fig.5. The request counts REQ (external or local) are entered from bus B into the delay element 71. In a preferred embodiment, the delay element 71 is also operated as a FIFO storage. The writing of request counts into the delay element is controlled by a timing signal derived (73) from the signals (slots) on bus B. The delay of the element 71, once it has been selected, stays constant. Request counts reaching the top of the delay element are read out under control of a timing signal derived (75) from the signals (slots) on bus A, and then inserted into the bottom position of the FIFO request queue 41. Thus, the access request counts are entered into and removed from the delay element at the pace of the slots on bus B and bus A, resepctively.

Initialization of the selective delay in each station is done as follows: The headend station inserts into a generated slot a first particular initialization marker M1 which propagates over the A bus and then the B bus to all stations. In each station recognizing the M1 marker, the B-clock timing signal for writing into the delay element 71 is started, and the request field contents of the slot carrying the M1 marker is inserted into the empty delay element 71, i.e. into its top position. The A-clock timing signal is inhibited during the initialization so that this first request count stays in the top position of the FIFO storage (delay element). Request counts of subsequent slots are written into the delay element which thus accumulates a number of such request counts (but all being zero). After a preselected time interval b corresponding to the system latency normalized to the slot transmission time (total propagation time, which depends on bus propagation speed, number of stations, etc.) the headend station inserts another particular initialization marker M2 into a generated slot which it releases on bus A. When this slot reaches the access point 67 of a station, it is detected by means 77 which then enable the A-clock timing signal (75) for reading out the count value in the top position of the delay element (FIFO) and transferring it into the FIFO request queue 41. The delay of element 71 (i.e. the number of request counts stored therein) now remains fixed. It is important that for each slot passing on bus B, a count is inserted into the delay element, even if it is zero, to thereby keep the number of request counts in the delay element and thus its delay constant.

It is further important that the propagation time on the two busses between points 67 and 69, including the selective delay of element 71, is an integer multiple of the slot time interval. This is possible due to the fact that two different timing signals (having the same frequency but independent phase) are used for writing and reading in the delay element, so that the selective delay can be a non-integer number of slot time intervals.

Fig.6 illustrates the resulting situation for two selected stations i and j. Again, the contents of the two request queues i and j are different as usual. The number of counts stored in the delay elements i and j are also different, reflecting the different individual delay of each station. The total delay or propagation time from point 67 to point 69 of each station is the same (= b) for all stations.

Another feature of the disclosed system is as follows (it will be shown only in the detailed implementation described in the next section): A sequence of external request counts (no local request occurred between them) need not be kept separate in the FIFO request queue, because they must anyway be served consecutively (as seen locally) before the respective station gains access to a slot for a pending local request. Thus, when a sequence of consecutive external requests (as well as a sequence of consecutive local requests) is entered into the FIFO request queue where this sequence would occupy several consecutive queue positions, they can be accumulated into a single request count, thus occupying only a single position. This results in shorter request queues requiring less storage. However, it should be noted that the accumulation (adding up of several requests from different slots) must only be made after these counts leave the delay element, because otherwise the once initialized delay would not be maintained.

4) Implementation of Request Queue and Selective Delay Element in a Station

A block diagram of the combined implementation of all the features described above, is shown in Fig.7 for one station. The circuitry is connected between a Bus A (31) and a Bus B (33) of a folded bus system (as shown in Fig.1B). Due to a signal conversion at each node (as will be described in connection with Fig.11) the signals on each bus at the node/station interface may be different from the signals on the connecting bus sections (other physical signal, other coding) but the information is the same. Therefore, in Fig.7 bus A and bus B carry reference numbers 31(X) and 33(X), respectively.

Two controllers (Fig.7) are connected to the two busses and to the other circuitry for controlling operation of the FIFO request queue circuitry. Bus A Controller 81 receives all signals from bus A; it is shown in more detail in Fig.8. Bus B Controller 83 receives all signals from bus B; it is shown in more detail in Fig.9.

A FIFO request queue storage 41, and a delay element 71 which is also implemented as a FIFO storage, are provided as already shown in Fig.5. Delay element 71 has e.g. 2048 positions each for holding an eight-bit request count value. FIFO request queue storage 41 has also 2048 positions, each for holding a 16-bit request count value. The request queue storage positions have a larger size than those of the delay element because count values from the delay element may be combined by addition before being entered into the request queue 41.

Queue Management 85 is connected to the FIFO request queue 41 for controlling its operation, and Delay Management 87 is connected to the delay element 71 for controlling its operation. Between the request queue FIFO and the delay element FIFO, there is provided an accumulator/ALU unit 89 (ACCU/ALU) with accumulator/ALU controller 90, for combining request counts read from delay element 71 before entering them into request queue 41. At the input of delay element 71 there is an input register 91. At the exit of the FIFO request queue 41, there is a 16-bit decrementer 93 which can hold one request count received from the request queue, and can decrease the current value by one unit when a respective control signal occurs.

Each of the positions of request queue FIFO 41 and of delay element FIFO 71, and also the input register 91 and the decrementer 93 have one extra bit position (41A, 71A, 91A, 93A) for holding one tag bit indicating whether the current contents in the respective position/register represents a local request count (LOC-TAG) or an external request count.

For transmitting data on bus A, a data segment buffer unit (DATA) 95 for holding a data segment waiting for transmission, and a data multiplexer 97 for overlaying data signals to the data stream on bus A are provided. Additional insert circuitry 99 in bus A allows to set the Busy/Free bit in a passing slot to "1" when data are to be inserted through the multiplexer 97.

The data segment buffer unit 95 preferably comprises two buffers, each of one data segment size, which are used alternatingly so that a segment can be read out to the bus from one of the twin buffers while the next segment (arriving from the station's higher layer) is inserted into the other one of the twin buffers.

It should be noted here that the segments transferred from the station to buffer unit 95 are just portions taken from a larger data packet to fit the buffer size and the slot segment size. They do not contain any count value or other housekeeping information to identify them separately. This is different from the segments usually provided in distributed queueing transmission systems; such segments must have some extra identifying information because they could be separated from the other segments of the same data packet, and thus must be reassembled at the destination by a special procedure using this extra

information.

For transmitting a request count on bus B, a local request register 101 for holding one local request count (LOC-REQ) waiting for transmission, and a request multiplexer 103 for overlaying request count signals to the stream on bus B are provided.

A certain delay is required between the points where signals are copied from each bus, and the points where data signals are inserted on the bus, because some test must be made on the copied data before a decision is made whether waiting data (or a local request count) are actually inserted into the passing slot. To ensure this minimum delay of one word, a delay register 105 is provided in bus A, and a delay register 107 in bus B.

This is basically all the circuitry which is required in each station for holding the request queues and maintaining a delay, in accordance with the invention. Interconnecting data lines and control signal lines will be identified individually in the following description of the operation of this circuitry.

Operation of Request Queue and Delay Element:

a) Insertion of Request Counts: Bus B Controller 83 watches the data stream on bus B and tests each request field in a passing slot whether it contains a zero count or an actual request count value. It receives a B-clock signal on line 111 from front-end circuitry (signal converters, as will be shown in Fig.11) to enable it to identify the fields in passing slots properly. If a zero count is detected (allowing insertion of a local request count into the passing ACF), then an Enable Multiplexer signal on line 113 is activated to enable the multiplexer 103 to insert the local request count waiting in register 101 via lines 115 into the passing ACF. Immediately thereafter, input register 91 is enabled by an Enable Register signal on line 117 to receive the contents of the passing request field, which now contains the just inserted local request count. Simultaneously, an active signal on line 119 (LOC-TAG) indicates that the count inserted into register 91 is a local request, thus establishing a local tag (LOC-TAG) with the entry in this register. Then, a credit signal is transferred on line 121 to the station main controller SMC giving it credit to send the next local request count into LOC-REQ register 101. When the SMC actually sends the next LOC-REQ on lines 123, it activates the signal on line 125 for enabling register 101 to receive the count and for notifying the Bus B Controller 83 that another local request is waiting for insertion.

If controller 83 detects that there is an actual external request count value in the request field of the passing ACF, then it does not activate the signals for local request insertion. However, the signal Enable Register on line 117 is activated each time a passing request field is seen, so that not only a transmitted local request count (as just described) is inserted into the input register, but also each external request count. However, the LOC-TAG will remain zero, thus indicating that the contents of register 91 is an external request. It is important (as will be explained later) that if the arriving (external) count is zero and no local request count is inserted (so that the request field in the passing slot remains empty), nevertheless a count is inserted into register 91, which will then be a zero count.

Bus B Controller 83 also furnishes a delay-in timing signal on line 127. This is a clock signal which is used by delay management 87, as will be explained later.

b) Insertion of Data: Bus A Controller 81 watches the data stream on bus A and tests the Busy/Free field in each passing slot. It receives an A clock signal on line 129 from front-end circuitry (signal converters, as will be shown in Fig.11) to enable it to identify the fields in passing slots properly. If it detects a free slot indication, it will activate the Decrement signal on line 131 to decrease the contents of decrementer 93 by one unit because now a pending slot request can be served (be it by the own station or another station downstream).

If the request count in decrementer 93 represents a local request, the tag in position 93A will be "1" and an indication (LOC-TAG) will be given on line 133 to the controller 81. Waiting local data can then be inserted into the passing slot. A Set Busy signal is activated on line 135 so that insert circuitry 99 converts the bit in the passing Busy/Free field. An Enable Multiplexer signal on line 137 is activated to enable multiplexer 97 to insert the local data segment waiting in data segment buffer unit 95, via lines 139 into the data segment field of the slot passing on bus A. Then, a Send Data signal on line 141 to the SMC is activated to give the SMC credit for sending another data segment. The SCM then sends the next data segment on lines 143.

As was already mentioned above, the data segment buffer unit 95 preferably includes twin buffers for alternating usage, to enable simultaneous reading of one and insertion of another data segment. As such twin buffer operation is well known, it need not to be described here in more detail.

Bus A Controller 81 also furnishes a delay-out timing signal on line 145. This is a clock signal which is used by delay management 87, as will be explained later.

c) Transfer of Request Counts Through Delay and Queue: Queue management 85 and delay management 87 control the writing of request counts into and the reading of request counts out of their respective associated FIFO storage. They keep pointers which indicate the current top position and the current bottom position in the respective FIFO storage.

A count is read into delay FIFO 71 from input register 91 for each passing slot on bus B, under control of the delay-in timing signal from controller 83. The tag in field 91A is also transferred to the delay FIFO. A request count is read from the top of delay FIFO 71 via data lines 146 into the ACCU/ALU unit 89 for each passing slot on bus A, under control of the delay-out timing signal from controller 81. The LOC-TAG is also transferred via line 147 to the ACCU/ALU unit 89 and to its controller 90. Controller 90 is notified by a signal on lines 148 of each transfer of a request count into ACCU/ALU unit 89.

The ACCU/ALU unit operates as follows: As long as the LOC-TAG value stays the same (e.g. as long as it is zero indicating external request counts), the ACCU/ALU unit adds the newly transferred count value to the accumulated value it already keeps in an accumulator register. As soon as the LOC-TAG value changes (e.g., if after a sequence of external request counts, a local request count with LOC-TAG = 1 is transferred from the delay FIFO to the ACCU/ALU unit), then the contents of the accumulator register is transferred through lines 149 into the FIFO request queue 41 as a single count, together with a respective LOC-TAG value. The queue management 85 is notified of the transfer by a control signal on lines 150, to insert the accumulated count value into its bottom position. The request count value just transferred from the delay FIFO 71 into ACCU/ALU unit 89 is kept there in the accumulator register.

Thus, consecutive request counts of the same kind (external or local) contained in the delay FIFO are combined to form a single request count value for the FIFO request queue, but the two kinds are still well distinguished in the FIFO request queue despite the combination.

Whenever the count contained in decrementer 93 reaches the value zero, a respective signal is activated on line 151, notifying the queue management that now the next request count value from the top position of the queue must be transferred into the decrementer 93 (together with its tag).

To avoid complete emptying of the FIFO request queue 41, the accumulation of request counts could be intermediately suspended in the following way: A threshold is stored in the queue management 85, indicating the minimum number of FIFO locations which should contain request counts. The queue management then constantly watches the number of filled FIFO locations (difference between input and output pointers) and compares it to the threshold. If this number falls below the threshold, a respective control signal (suspend accumulation) is sent via lines 150 to the ACCU/ALU controller 90 which thereafter no longer accumulates counts but transfers them individually as received from the delay unit 71. When the queue management detects that the number of filled request queue FIFO locations is again above the set threshold, then it sends another control signal (resume accumulation) to ACCU/ALU controller 90 which from then on again accumulates the request counts as described above.

d) Initialization of Delay: At the beginning of system operation, or when an automatic restart is necessary after a system failure, all bus controllers have stopped their delay-in and delay-out timing signals on lines 127 and 147, and the delay FIFOs (71) are reset. As already briefly explained in section 3, the headend station then releases two delay initialization markers, M1 and M2. The headend station continuously generates slots, and inserts the two markers in the ACFs of two slots which have a particular time distance (= b time slots) from each other. When the first marker M1 arrives at Bus B Controller 83 in a station, the controller starts furnishing the delay-in timing signal on line 127. Thus, from that time on request counts are written into the delay FIFO 71 (all having a value of zero). During this initialization time, no counts are read out from the delay FIFO, i.e. the delay-out signal is still inactive. As soon as the second marker M2 arrives at Bus A Controller 81, it starts furnishing the delay-out timing signal on line 147. In the meantime, a number of request counts have accumulated in the delay FIFO 71. This number will then remain constant during operation and determines the station's individual delay which compensates any propagation differences, as explained in section 3.

5) Details of Bus FIFO Controllers

Some details of the two controllers 81 and 83 will now be described with reference to Figs.8 and 9.

Fig. 8 is a block diagram of the essential portions of the Bus A Controller (81). It comprises means 153 for extracting from the data stream it receives from bus A, and under control of the clock signal (129) derived from the signals on bus A, the Access Control Field ACF of each passing slot and for transferring certain subfields to a marker M2 detector 154 and a Busy/Free detector 155. A Delay Set latch 157 is reset at each system start or restart. When a marker M2 is detected, latch 157 is set by a signal on line 159 and then enables by its output signal on line 161 a gate 163, for transferring the A-clock signal from line 129 to

the delay-out timing signal line 147.

When the Busy/Free detector 155 detects a zero in the B/F subfield of a passing ACF, then it activates at its output a control signal indicating that a free slot arrived which signal is transferred as Decrement signal over line 131 to decrementer 93. If the count in the decrementer represents a local request, as indicated by an active signal on LOC-TAG line 133, an AND gate 165 transfers the Slot Free signal on line 135 as Set Busy signal, for occupying the passing slot for a local data transfer. With a certain delay introduced by delay element 167, the output signal of AND gate 165 activates the Enable Multiplexer signal on line 137 for actually inserting the data into the passing slot, and the Send Data signal on line 141 for allowing the transfer of another local data segment.

Fig.9 is a block diagram of the essential portions of the Bus B Controller (83). It comprises means 173 for extracting from the data stream it receives from bus B, and under control of the clock signal (111) derived from the signals on bus B, the Access Control Field ACF of each passing slot and for transferring certain subfields to a marker M1 detector 175, a Request Field detector 177, and a Request Count = 0 (RQ = 0) detector 179. A Delay Set latch 181 is reset at each system start or restart. When a marker M1 is detected, latch 181 is set by a signal on line 183 and then enables by its output signal on line 185 a gate 187, for transferring the B-clock signal from line 111 to the delay-in timing signal line 127.

A Request Waiting latch 189 is reset at each system start, and when the controller sends a credit for a new request count by a signal on line 121. Data Waiting latch 189 is set by the New Request notification signal on line 125 when the SMC transfers a new local request count into the local request register 101. When the RQ = 0 detector 179 detects a zero in the request subfield of a passing ACF, then it activates at its output a control signal which is transferred to one input of an AND gate 191. If simultaneously the output signal of latch 189 on line 193 indicates that a local request is waiting for transmission, then the output signal of AND gate 191 activates on line 113 the Enable Multiplexer signal for multiplexer 103 to actually insert the waiting local request count into the request subfield of the passing slot. With a certain delay introduced by a delay element 195, the output signal of AND gate 191 activates the local tag (LOC-TAG) signal on line 119 for inserting the local tag into the tag portion 91A of input register 91, and the Credit signal on line 121 for allowing the transfer of another local request from the SMC. This signal also resets the Request Waiting latch.

Whenever a request field is detected on bus B (be its contents zero or not), detector 177 activates on its output a respective control signal which is delayed by a delay element 197, and then transferred as Enable Register signal on line 117 to the input register 91 for enabling it to copy the request count from the slot which passes by on bus B.

6) Representation of Initialization Markers

As was described above, the headend station must insert two different markers into the ACF field of slots when the individual delay of the request queue in each station is to be initialized. The Reserved subfield (shown in the ACF format in Fig.2C) can be used for representing the markers M1 and M2. Below are shown examples for the marker representation.

```
(A)   R1   R2   R3   R4
      ---------------

       1    0    X    X   =   M1
       1    1    X    X   =   M2

       0    X    X    X   =   other


(B)   R1   R2   R3   R4
      ---------------

       1    0    0    X   =   M1  (original)
       1    0    1    X   =   M1  (returning)
       1    1    X    X   =   M2

       0    X    X    X   =   other
```

For the folded bus system (Fig.1B) which was used as the example for above described embodiment, the respective representation is shown at (A). The four bits of the Reserved subfield are designated as R1,

R2, R3, and R4. To indicate the presence of a marker, R1 is set to 1. The two markers are distinguished by bit R2. The remaining two bits R3 and R4 are still available for other purposes. As was described above, marker M1 is used on bus B whereto it automatically propagates in a folded bus system. Marker M2 is only used on bus A.

For a dual bus system as shown in Fig.1A, the situation is slightly different because usually slots propagate only along one bus and do not return on the other bus. Therefore, a modification is necessary to allow the M1 markers when they have reached the end of one bus, to be further propagated on the other bus. An additional representation is required to distinguish an M1 marker on its original bus from an M1 marker returning on the other bus. The third reserved bit R3 is used for this purpose. The representation is shown above at (B).

7) Modification in Headend Stations of Dual Bus System for Delay Initialization

As indicated above, the M1 markers must be wrapped around to the other bus in a dual bus system which requires some modification in the headend stations. This is shown in Fig.10 (which is the modified form of the dual bus network shown in Fig.1A). The headend station 15 for bus A (HE-A) comprises as usual a generator/sender 201 which generates the slots and inserts some information into the ACF field when transmitting them. Also, headend station HE-B (17) comprises such generator/sender 203 for slots. Headend station HE-A generates for initialization, markers M1(A) and M2(A). The first one propagates along bus A but must also propagate along bus B because it has to be detected there. Therefore, bus section 21 (which could be omitted in a normal dual bus system) must be included so that each slot of bus A can propagate through to headend station HE-B. An extra receiver 205 is provided in HE-B which receives the slots and can detect the presence of an original marker M1 in an ACF. If it detects one, it sends a control signal on line 207 to slot generator/sender 203, which then inserts a returning marker M1' into the next slot it issues on bus B. This marker can then be recognized by all stations to start their delay-in timing signal.

In headend station HE-A, there is also provided an additional receiver 209 (receiving the slots from bus B on section 23), and a control signal line 211. Thus, also headend station HE-A can recognize an original M1 marker (in this case one that was issued be headend HE-B), and cause its generator/sender 201 to insert a returning marker M1' into a slot propagating along its bus A. The receivers (205, 209) in both headend stations ignore returning M1' markers, as well as M2 markers; thus, a returning marker M1' propagates only once along the counter-bus, and a marker M2 propagates only once along its original bus.

8) Duplication for Dual Bus Network

In a dual bus network as shown in Fig.1A, there are actually two independent request and access arrangements: Request for data transmission access on bus A are inserted and propagated on bus B, and requests for data transmission access on bus B are inserted and propagated on bus A. While the circuitry shown and described above was designed for a folded bus system in which requests are only inserted and transmitted on the inbound bus, it is clear (and was already mentioned) that the invented access technique can also be used in dual bus systems, but all the circuitry (request queue FIFO and delay FIFO) must then be duplicated.

9) Signal Conversion between Busses and Stations

Fig.11 is a schematic representation of the front-end circuitry which provides signal conversion and appropriate code conversion between system busses and station circuitry. Typically, information could be transferred on the busses (31, 33) between stations in the form of bit-serial optical signals. Within station circuitry, information would be represented as electrical signals in word-parallel form.

Incoming bus A (31) is connected to Bus A Front-End Receiver 215 (including an optical/electrical signal converter) which furnishes on its parallel output bus lines 31(X) the data, and on a clock line 129 the regenerated A-clock signal, both in electrical form, to the station's bus access circuitry 217. Data generated in the station to be further propagated on the bus are furnished on parallel bus lines 31(X) to Bus A Front-End Transmitter 219 (including an electrical/optical signal converter) which transmits them in converted form as optical serial signal on outgoing bus A (31).

Similarly, bus B (33) is connected through Bus B Front-End Receiver 221 (including an optical/electrical signal converter) and through Bus B Front-End Transmitter 223 (including an electrical/optical signal converter) to the station's bus access circuitry 217. A B-clock signal as derived by front-end receiver 221 from data on bus B is furnished on line 111.

The station's other equipment (higher layer) including the Station Main Controller SMC, all indicated as one block 225 in Fig.11, is connected to the bus access circuitry 217 by several data and control lines for the exchange of data and control signals.

Those portions of bus access circuitry 217 which are essential for present invention were shown in Figs.7, 8, and 9 and explained in the respective text. It should be noted that bus access circuitry, besides receiving and transmitting access requests on bus B, and transmitting data on bus A (all explained with reference to Figs.7, 8, 9) also receives data from bus B which are directed (addressed) to the station. However, data reception is not relevant for the invention and therefore not shown here in detail.

10) Arrangement for Multiple Priorities

The circuitry shown in the above description will handle all requests for one priority class. If no different priorities are provided, this is sufficient for the operation. However, if multiple priorities are introduced as was mentioned in section 1 with the description of Fig.2A - 2D (slot format and ACF format), i.e. if separate requests have to be made for the various priorities and separate slots are issued, the request queue with selective delay and associated circuitry have to be provided as many times as there are priority classes. Thus, when four classes are introduced, each station must have four request queue FIFO stores.

The delay element FIFO could also be provided in multiple form, one for each priority. However, a single shared delay element FIFO would be sufficient if each entry is amended by a priority tag (two bits for four priorities). At the delay element output, a demultiplexer must then be provided for distributing the request counts read from the delay element FIFO into the multiple request queues (e.g. four request queues) according to the priority tag.

Claims

1. Method of regulating access to a communication system with two counterflowing transmission busses and a plurality of stations connected between them, sequential time slots being generated on a first one of said busses for data transmission, each station transmitting access requests for slots on a respective second one of said busses determining said station's right to transmit data in a slot on said first bus on the basis of external access requests it has detected from other stations and own local access requests;
   said method comprising the step of:
   - providing, in said access requests transmitted on said second bus, for a reservation of multiple slots by a station,

   and the following steps to be executed in each station:
   - maintaining an access request queue (41) of said external and local access requests reflecting their temporal relationship;
   - entering a local request in said access request queue upon transmitting it on the second bus;
   - providing a station-individual delay (71) for access requests to be entered into the access request queue to compensate for propagation time differences existing for time slots and access requests with respect to the different stations of the system.

2. Method in accordance with claim 1, further comprising the steps of:
   - storing said access requests as numbers, each representing either an external request count or a local request count, sequentially in one FIFO request storage (41);
   - transferring the contents of the top position of said FIFO request storage into a decrementing register (93);
   - decrementing the contents of said decrementing register for each empty slot detected on said first bus by the station; and
   - when the contents of said decrementing register has reached the value zero, transferring the contents from the next position of said FIFO request storage into said decrementing register.

3. Method in accordance with claim 1, further comprising the step of:
   - providing said station-individual delay by storing access requests, prior to entering them into said access request queue (41), in a FIFO delay storage (71) containing a preselected number of entries determining the amount of delay.

4. Method in accordance with claim 3, further comprising the steps of:

- initializing the delay effected by said FIFO delay storage in each station, by transmitting on said first bus two initializing markers (M1, M2) which are separated by a preselected time interval (b);
- starting, in any station when it detects the first marker on the second bus, a first timing signal (127) derived from slots on said second bus for controlling writing of access requests into said FIFO delay storage, and entering one entry into said FIFO delay storage for each slot passing on said second bus, which entry either represents a request count contained in a slot or is equal to zero if a slot contains no access request; and
- starting, in any station when it detects the second marker on the first bus, a second timing signal (145) derived from slots on said first bus for controlling the reading of one entry from said FIFO delay storage per slot on said first bus to transfer it into said access request queue.

5. Method in accordance with claim 4, in a system which comprises two separate busses, and in which slots propagate only along the bus on which they are generated, comprising the step of:
- transferring the first initialization marker (M1), when it has reached the end of the one bus on which it was released, in its original or in modified form to the other bus so that it can propagate also along said other bus.

6. Method in accordance with claim 3, further comprising the step of:
- accumulating (89, 90) consecutive external access requests which are read from said FIFO delay storage, and entering them as a single access request count into said access request queue.

7. Method in accordance with claim 1, in a system providing a plurality of priorities for requesting access to slots, comprising the steps of:
- generating each of said time slots with a priority indication;
- inserting an access request for a given priority only into a slot carrying the respective priority indication; and
- providing a separate access request queue for each of said priorities for receiving external access requests and local access requests of the respective priority.

8. Method in accordance with claim 1, further comprising the steps of:
- transmitting a local access request by inserting it into a slot on the second one of said busses, and
- copying each access request seen in a passing slot on said second bus, including any local access request just transmitted by the respective station, for insertion into the access request queue of the respective station.

9. Method of regulating access to a communication network with two unidirectional counterflowing transmission busses, a plurality of stations each connected to both busses, and at least one headend station for generating time slots propagating on said busses, each slot having an access control field (ACF) and a data segment field, in which method each station requests slot access by inserting an access request into a request subfield of said access control field in a passing slot, and determines its right to access a slot for data transmission on the basis of its local access requests and the external access requests it has detected from other stations in passing slots,
said method comprising the step of:
- providing, in said request subfield, for a reservation of multiple slots;

and comprising the following steps to be executed in each station:
- maintaining a record of local access requests (LOC-REQ) generated by the respective station and of external access requests (EXT-REQ) seen from other stations in the form of a FIFO request queue (41), said record reflecting the sequential order of occurrence of said access requests;
- inserting a request count for each local access request for one or multiple slots into an empty request subfield of a passing slot,
- thereafter inserting a respective local request count into the FIFO request queue; and
- monitoring the contents of the request subfield in each passing slot, and copying any external access request and forwarding (91) it as external request count for insertion into the FIFO request queue.

10. Method in accordance with claim 9, further comprising the steps of:

- decrementing the access request count in the top position of the request queue in each station when it detects a free slot at its input to remove one specific slot request; and
- introducing initially a station-individual delay (71) at each FIFO request queue such that, when a specific slot request is about to be removed from the request queue in a station, the respective station detects passing at its input the same specific slot which was or will be detected by any other station at its input when the same specific slot request is about to be removed from the request queue in said other station.

11. Method in accordance with claim 10, further comprising the step of:

- providing said station-individual delay by storing access requests, prior to entering them into said FIFO request queue, in a FIFO delay storage (71) containing a preselected number of entries determining the amount of delay.

12. Method in accordance with claim 11, further comprising the steps of:

- initializing the delay effected by said FIFO delay storage in each station by transmitting on a first one of said busses two initializing markers (M1, M2) which are separated by a preselected time interval (b);
- starting, in any station when it detects the first marker on the respective second one of said busses, a first timing signal (127) derived from slots on said second bus for controlling writing of request counts into said FIFO delay storage, and entering one entry into said FIFO delay storage for each slot passing on said second bus, which entry either represents a request count contained in a slot or is equal to zero if a slot contains no access request; and
- starting, in any station when it detects the second marker on the first bus, a second timing signal (145) derived from slots on said first bus, for controlling the reading of one entry from said FIFO delay storage per slot on the first bus, to transfer it into said FIFO request queue.

13. Method in accordance with claim 10, further comprising the step of:

- accumulating (89, 90) external access requests which are read consecutively from said FIFO delay storage without an intervening local access request, prior to entering them into said FIFO request queue, and also accumulating local access requests which are read from said FIFO delay storage without an intervening external access request, prior to entering them into said FIFO request queue.

14. A communication network with two unidirectional counterflowing transmission busses, a plurality of stations each connected to both busses, and at least one headend station for generating time slots on one of said busses, in which network each station requests slot access by inserting an access request into a passing slot, keeps a record of pending access requests, and determines its right to access a free slot for data transmission on the basis of its own local access requests and of external access requests it has detected from other stations in passing slots, each said slot access request including an indication of the number of slots requested,
said network including a distributed queue multiple access apparatus in each said station which comprises:

- request queue storage means (41, 85) for storing in sequential order request count values representing external access requests (EXT-REQ) and local access requests (LOC-REQ), respectively,
- delay means (71, 87) connected in series with said request queue storage means for providing, individually for the respective station, a selective delay for request count values being entered into said request queue storage means, for compensating propagation delay differences which exist for slots with respect to the different stations.
- inserting means (103, 113, 115; 91, 117, 119) for inserting a local request count value into a passing slot on one of said busses, and for inserting the same local request count value into said delay means (71, 87) on said request queue storage means (41, 85) upon insertion of said same local request count value into a passing slot on said one bus.

15. Apparatus in accordance with claim 14, in which said request queue storage means (41, 85) is a FIFO storage from which always the oldest stored request count value is read out first.

16. Apparatus in accordance with claim 14, further comprising decrementing means (93) for receiving a request count value read from said request queue storage means (41, 85) and for decrementing its contents by one unit upon occurrence of a respective control signal (131) issued by a bus controller (81) which detects the occurence of empty slots, and for furnishing another control signal (151) to said request queue storage means when the contents of said decrementing means has become zero, to initiate the transfer of another request count value.

17. Apparatus in accordance with claim 14, in which said delay means (41, 87) is a FIFO delay storage (41, 87) from which always the oldest stored request count value is read out first.

18. Apparatus in accordance with claim 17, further comprising means (111, 175, 181, 183, 185, 187) for deriving a first timing signal (127) for controlling the writing of request count values into said FIFO delay storage (71), from the slots on one (B) of said two busses, and means (129, 154, 157, 159, 161, 163) for deriving a second timing signal (147) for controlling the reading of request count values out of said FIFO delay storage, from the slots on the other one (A) of said two busses, such that the writing into and reading from said FIFO delay storage occurs at the pace of the slots on the respective busses.

19. Apparatus in accordance with claim 14 or 18, further having means for initializing the delay of said delay means (71, 87) in response to two different initialization markers (M1, M2) which are transmitted by said headend station with a given time distance between them; said initialization means comprising:
- means (111, 175, 181, 183, 185, 187) for detecting on one (B) of said busses a first one of said markers (M1, M1') and for starting, upon detection of said first marker, a first timing signal (127) for writing request count values into said delay means (71, 87), and
- means (129, 154, 157, 159, 161, 163) for detecting on the other one (A) of said busses the second one of said markers (M2) and for starting, upon detection of said second marker, a second timing signal (147) for reading request count values from said delay means for insertion into said request queue storage means (41, 85).

20. Apparatus in accordance with claim 14, further comprising means (89,90) for accumulating by addition sequential external request count values on one hand and sequential local request count values on the other hand, which were read from said delay means (71, 87), prior to entering them into said request queue storage means (41, 85).

21. Apparatus in accordance with claim 14, for use in a system in which plural priority categories are provided for access requests; the apparatus comprising as many parallel request queue storage means and associated circuitry as the system has priority categories, so that access requests can be stored and handled separately for each one of the priority categories.

Patentansprüche

1. Verfahren zur Regulierung des Zugriffs auf ein Kommunikationssystem mit zwei gegenläufigen Übertragungsbussen und einer Vielzahl von Stationen, die zwischen ihnen angeschlossen sind, wobei sequentielle Zeitschlitze auf einem ersten der beiden Busse zur Datenübertragung generiert werden, jede Station Zugriffsanforderungen für Schlitze auf einem entsprechenden zweiten der beiden Busse überträgt, wobei die Berechtigung einer Station, Daten in einem Schlitz auf dem ersten Bus zu übertragen, auf der Grundlage von externen Zugriffsanforderungen, die sie von anderen Stationen wahrgenommen hat, sowie eigener lokaler Zugriffsanforderungen bestimmt wird;
wobei das Verfahren folgenden Schritt umfaßt:
- dafür zu sorgen, daß eine Station in den auf dem zweiten Bus übertragenen Zugriffsanforderungen eine Reservierung von mehreren Schlitzen vornehmen kann,

und die folgenden Schritte in jeder Station durchzuführen sind:
- Verwalten einer Zugriffs-Anforderungswarteschlange (41) der externen und lokalen Zugriffsanforderungen, die ihre zeitliche Beziehung widerspiegelt;
- Eintragen einer lokalen Anforderung in die Zugriffs-Anforderungswarteschlange im Anschluß an deren Übertragung auf dem zweiten Bus;
- Vorsehen einer stationsspezifischen Verzögerung (71) für Zugriffsanforderungen, die in die Zugriffs-Anforderungswarteschlange eingetragen werden sollen, um unterschiedliche Laufzeiten auszugleichen, die es bei Zeitschlitzen und Zugriffsanforderungen hinsichtlich der verschiedenen

Stationen des Systems gibt.

2. Das Verfahren gemäß Anspruch 1, das ferner folgende Schritte umfaßt:
   - Sequentielles Speichern der Zugriffsanforderungen in einem FIFO-Anforderungsspeicher (41) als Zahlen, wobei jede Zahl entweder eine Anzahl externer Anforderungen oder eine Anzahl lokaler Anforderungen darstellt;
   - Übertragen des Inhalts des obersten Speicherplatzes des FIFO-Anforderungsspeichers in ein Dekrementierungsregister (93);
   - Dekrementieren des Inhalts des Dekrementierungsregisters für jeden leeren Schlitz, der von der Station auf dem ersten Bus festgestellt wurde, und,
   - sobald der Inhalt des Dekrementierungsregisters den Wert Null erreicht hat, Übertragen des Inhalts des nächsten Speicherplatzes des FIFO-Anforderungsspeichers in das Dekrementierungsregister.

3. Das Verfahren gemäß Anspruch 1, das ferner folgenden Schritt umfaßt:
   - Vorsehen der stationsspezifischen Verzögerung, indem Zugriffsanforderungen, bevor sie in die Zugriffs-Anforderungswarteschlange (41) eingetragen werden, in einem FIFO-Verzögerungsspeicher (71) gespeichert werden, der eine vorher ausgewählte Anzahl von Einträgen enthält, welche die Verzögerungszeit bestimmen.

4. Das Verfahren gemäß Anspruch 3, das ferner folgende Schritte umfaßt:
   - Initialisieren der vom FIFO-Verzögerungsspeicher vorgenommenen Verzögerung in jeder Station, indem auf dem ersten Bus zwei Initialisierungsmarken (M1, M2) übertragen werden, die durch ein vorher ausgewähltes Zeitintervall (b) getrennt sind;
   - Auslösen eines ersten, von Schlitzen auf dem zweiten Bus abgeleiteten Zeitsignals (127) in einer beliebigen Station, wenn sie die erste Marke auf dem zweiten Bus erkennt, um das Schreiben von Zugriffsanforderungen in den FIFO-Verzögerungsspeicher zu steuern, und um für jeden Schlitz, der auf dem zweiten Bus vorbeiläuft, eine Eingabe in den FIFO-Verzögerungsspeicher vorzunehmen, wobei diese Eingabe entweder eine in einem Schlitz enthaltene Anforderungsanzahl darstellt oder gleich Null ist, wenn ein Schlitz keine Zugriffsanforderung enthält, und
   - Auslösen eines zweiten, von Schlitzen auf dem ersten Bus abgeleiteten Zeitsignals (145) in einer beliebigen Station, wenn sie die zweite Marke auf dem ersten Bus erkennt, um das Lesen einer Eingabe aus dem FIFO-Verzögerungsspeicher je Schlitz auf dem ersten Bus zu steuern, um sie in die Zugriffs-Anforderungswarteschlange zu übertragen.

5. Das Verfahren gemäß Anspruch 4, das in einem System mit zwei getrennten Bussen und in dem Schlitze nur auf dem Bus entlanglaufen, auf dem sie generiert werden, folgenden Schritt umfaßt:
   - Übertragen der ersten Initialisierungsmarke (M1) in ihrer ursprünglichen oder in modifizierter Form auf den anderen Bus, wenn sie am Ende des einen Busses angekommen ist, auf dem sie freigesetzt wurde, damit sie auch entlang des anderen Bus laufen kann.

6. Das Verfahren gemäß Anspruch 3, das ferner folgenden Schritt umfaßt:
   - Summieren (89, 90) fortlaufender externer Zugriffsanforderungen, die aus dem FIFO-Verzögerungsspeicher gelesen werden, um sie dann als eine einzige Zugriffsanforderungsanzahl in die Zugriffs-Anforderungswarteschlange einzutragen.

7. Das Verfahren gemäß Anspruch 1, das in einem System, in dem eine Vielzahl von Prioritäten für Zugriffsanforderungen für Schlitze vorgesehen ist, folgende Schritte umfaßt:
   - Generieren eines jeden Zeitschlitzes mit einem Prioritätshinweis;
   - Einfügen einer Zugriffsanforderung für eine bestimmte Priorität nur in einen Schlitz, der den entsprechenden Prioritätshinweis trägt, und
   - Vorsehen einer getrennten Zugriffs-Anforderungswarteschlange für jede einzelne der Prioritäten, um externe Zugriffsanforderungen und lokale Zugriffsanforderungen der entsprechenden Priorität zu empfangen.

8. Das Verfahren gemäß Anspruch 1, das ferner folgende Schritte umfaßt:
   - Übertragen einer lokalen Zugriffsanforderung, indem sie in einen Schlitz auf dem zweiten der beiden Busse eingefügt wird, und

- Kopieren einer jeden Zugriffsanforderung, die in einem passierenden Schlitz auf dem zweiten Bus festgestellt wurde, einschließlich jedweder lokalen Zugriffsanforderung, die von der entsprechenden Station gerade übertragen wurde, um sie in die Zugriffs-Anforderungswarteschlange der entsprechenden Station einzufügen.

9. Verfahren zur Regulierung des Zugriffs auf ein Kommunikationsnetzwerk mit zwei gegenläufigen Einweg-Übertragungsbussen, einer Vielzahl von Stationen, von denen jede an beide Busse angeschlossen ist, und mindestens einer Kopfstation zur Generierung von Zeitschlitzen, die auf den Bussen laufen, wobei jeder Schlitz ein Zugriffskontrollfeld (ACF) und ein Datensegmentfeld hat, wobei gemäss dem Verfahren eine jede Station den Zugriff auf einen Schlitz anfordert, indem sie eine Zugriffsanforderung in ein Anforderungsteilfeld des Zugriffskontrollfeldes in einem passierenden Schlitz einfügt, und ihre Berechtigung, auf einen Schlitz zur Datenübertragung zuzugreifen, auf der Grundlage ihrer lokalen Zugriffsanforderungen sowie der externen Zugriffsanforderungen bestimmt, die sie von anderen Stationen in passierenden Schlitzen festgestellt hat,
wobei das Verfahren folgenden Schritt umfaßt:

- dafür zu sorgen, daß eine Reservierung mehrerer Schlitze in dem Anforderungsteilfeld vorgenommen werden kann,

sowie die folgenden Schritte umfaßt, die in jeder Station durchzuführen sind:

- Verwalten eines Verzeichnisses lokaler Zugriffsanforderungen (LOC-REQ), die von der entsprechenden Station generiert wurden, sowie externer Zugriffsanforderungen (EXT-REQ), die sie von anderen Stationen wahrgenommen hat, in Form einer FIFO-Anforderungswarteschlange (41), wobei das Verzeichnis die sequentielle Reihenfolge widerspiegelt, in der die Zugriffsanforderungen aufgetreten sind;
- Einfügen einer Anforderungsanzahl für jede lokale Zugriffsanforderung für einen oder mehrere Schlitze in ein leeres Anforderungsteilfeld eines passierenden Schlitzes,
- anschließend Einfügen einer entsprechenden Anzahl lokaler Anforderungen in die FIFO-Anforderungswarteschlange und
- Überwachen des Inhalts des Anforderungsteilfeldes in jedem passierenden Schlitz und Kopieren jedweder externen Zugriffsanforderung sowie deren Weiterleitung (91) als Anzahl externer Anforderungen, um sie in die FIFO-Anforderungswarteschlange einzufügen.

10. Das Verfahren gemäß Anspruch 9, das ferner folgende Schritte umfaßt:

- Erniedrigen der Anzahl der Zugriffsanforderungen im obersten Speicherplatz der Anforderungswarteschlange in jeder Station, wenn sie einen freien Schlitz an ihrem Eingang feststellt, um eine bestimmte Anforderung für einen Schlitz zu entfernen; und
- anfängliches Einführen einer stationsspezifischen Verzögerung (71) bei jeder FIFO-Anforderungswarteschlange, und zwar derart, daß wenn eine Anforderung für einen bestimmten Schlitz kurz davor steht, aus der Anforderungswarteschlange einer Station entfernt zu werden, die entsprechende Station feststellt, daß derselbe bestimmte Schlitz gerade an ihrem Eingang vorbeiläuft, der von jeder beliebigen anderen Station an deren Eingang festgestellt wurde oder festgestellt wird, wenn die Anforderung für denselben bestimmten Schlitz kurz davor steht, aus der Anforderungswarteschlange der betreffenden anderen Station entfernt zu werden.

11. Das Verfahren gemäß Anspruch 10, das ferner folgenden Schritt umfaßt:

- Vorsehen der stationsspezifischen Verzögerung, indem Zugriffsanforderungen, bevor sie in die FIFO-Anforderungswarteschlange eingetragen werden, in einem FIFO-Verzögerungsspeicher (71) gespeichert werden, der eine vorher ausgewählte Anzahl von Eingaben enthält, welche die Verzögerungszeit bestimmen.

12. Das Verfahren gemäß Anspruch 11, das ferner folgende Schritte umfaßt:

- Initialisieren der vom FIFO-Verzögerungsspeicher vorgenommenen Verzögerung in jeder Station, indem auf einem ersten der beiden Busse zwei Initialisierungsmarken (M1, M2) übertragen werden, die durch ein vorher ausgewähltes Zeitintervall (b) getrennt sind;
- Auslösen eines ersten, von Schlitzen auf dem zweiten Bus abgeleiteten Zeitsignals (127) in einer beliebigen Station, wenn sie die erste Marke auf dem entsprechenden zweiten Bus erkennt, um das Schreiben von Anforderungsergebnissen in den FIFO-Verzögerungsspeicher zu steuern, und um für jeden Schlitz, der auf dem zweiten Bus vorbeiläuft, eine Eingabe in den FIFO-Verzögerungsspeicher vorzunehmen, wobei diese Eingabe entweder eine in einem Schlitz enthaltene

Anforderungsanzahl darstellt oder gleich Null ist, wenn ein Schlitz keine Zugriffsanforderung enthält, und

- Auslösen eines zweiten, von Schlitzen auf dem ersten Bus abgeleiteten Zeitsignals (145) in einer beliebigen Station, wenn sie die zweite Marke auf dem ersten Bus erkennt, um das Lesen einer Eingabe aus dem FIFO-Verzögerungsspeicher je Schlitz auf dem ersten Bus zu steuern, um sie in die FIFO-Anforderungswarteschlange zu übertragen.

13. Das Verfahren gemäß Anspruch 10, das ferner folgenden Schritt umfaßt:

- Summieren (89, 90) von externen Zugriffsanforderungen, die fortlaufend aus dem FIFO-Verzögerungsspeicher, ohne einer dazwischen auftretenden lokalen Zugriffsanforderung, gelesen werden, bevor sie in die FIFO-Anforderungswarteschlange eingetragen werden, ebenso wie Summieren von lokalen Zugriffsanforderungen, die aus dem FIFO-Verzögerungsspeicher, ohne einer dazwischen auftretenden externen Zugriffsanforderung, gelesen werden, bevor sie in die FIFO-Anforderungswarteschlange eingetragen werden.

14. Ein Kommunikationsnetzwerk mit zwei gegenläufigen Einweg-Übertragungsbussen, einer Vielzahl von Stationen, von denen jede mit beiden Bussen verbunden ist, und mindestens einer Kopfstation zur Generierung von Zeitschlitzen auf einem der Busse, wobei jede Station in dem Netzwerk den Zugriff auf einen Schlitz anfordert, indem sie eine Zugriffsanforderung in einen passierenden Schlitz einfügt, ein Verzeichnis der anstehenden Zugriffsanforderungen führt und ihre Berechtigung, auf einen freien Schlitz zur Datenübertragung zuzugreifen, auf der Grundlage ihrer eigenen lokalen Zugriffsanforderungen sowie externer Zugriffsanforderungen, die sie von anderen Stationen in passierenden Schlitzen festgestellt hat, bestimmt, und jede Zugriffsanforderung für Schlitze einen Hinweis auf die Anzahl der angeforderten Schlitze enthält,
wobei das Netzwerk in jeder Station eine Vorrichtung zum Vielfachzugriff mit verteilten Warteschlangen enthält, die folgendes aufweist:

- Anforderungswarteschlangen-Speichermittel (41, 85), um in sequentieller Reihenfolge Anforderungszählwerte zu speichern, die externe Zugriffsanforderungen (EXT-REQ) beziehungsweise lokale Zugriffsanforderungen (LOC-REQ) darstellen,
- Verzögerungsmittel (71, 87), die mit den Anforderungswarteschlangen-Speichermitteln in Reihe geschaltet sind, um eine selektive, stationsspezifische Verzögerung für Anforderungszählwerte vorzusehen, die in die Anforderungswarteschlangen-Speichermittel eingegeben werden, um unterschiedliche Laufzeitverzögerungen auszugleichen, die es bei Schlitzen hinsichtlich der verschiedenen Stationen gibt.
- Einfügemittel (103, 113, 115; 91, 117, 119), um einen Zählwert lokaler Anforderungen in einen auf einem der Busse passierenden Schlitz einzufügen, und um denselben Zählwert lokaler Anforderungen in die Verzögerungsmittel (71, 87) der Anforderungswarteschlangen-Speichermittel (41, 85) einzufügen, nachdem derselbe Zählwert lokaler Anforderungen in einen passierenden Schlitz auf dem einen Bus eingefügt wurde.

15. Die Vorrichtung gemäß Anspruch 14, in der es sich bei den Anforderungswarteschlangen-Speichermitteln (41, 85) um einen FIFO-Speicher handelt, aus dem immer der bereits am längsten gespeicherte Anforderungszählwert zuerst gelesen wird.

16. Die Vorrichtung gemäß Anspruch 14, die ferner Dekrementierungsmittel (93) aufweist, um einen Anforderungszählwert zu empfangen, der aus den Anforderungswarteschlangen-Speichermitteln (41, 85) gelesen wurde, und um ihren Inhalt nach dem Auftreten eines entsprechenden Steuersignals (131) um eine Einheit zu dekrementieren, das von einer Bussteuereinheit (81) ausgegeben wurde, die das Auftreten von leeren Schlitzen feststellt, und um ein weiteres Steuersignal (151) an die Anforderungswarteschlangen-Speichermittel zu übertragen, wenn der Inhalt der Dekrementierungsmittel auf Null zurückgegangen ist, um die Übertragung eines weiteren Anforderungszählwertes auszulösen.

17. Die Vorrichtung gemäß Anspruch 14, in der es sich bei den Verzögerungsmitteln (41, 87) um einen FIFO-Verzögerungsspeicher (41, 87) handelt, aus dem immer der bereits am längsten gespeicherte Anforderungszählwert zuerst gelesen wird.

18. Die Vorrichtung gemäß Anspruch 17, die ferner Mittel (111, 175, 181, 183, 185, 187) aufweist, um ein erstes Zeitsignal (127) von den Schlitzen auf einem (B) der beiden Busse abzuleiten, das dazu dient,

das Schreiben von Anforderungszählwerten in den FIFO-Verzögerungsspeicher (71) zu steuern, sowie Mittel (129, 154, 157, 159, 161, 163), um ein zweites Zeitsignal (147) von den Schlitzen auf dem anderen (A) der beiden Busse abzuleiten, das dazu dient, das Lesen von Anforderungszählwerten aus dem FIFO-Verzögerungsspeicher zu steuern, und zwar derart, daß das Schreiben in und das Lesen aus dem FIFO-Verzögerungsspeicher mit der Geschwindigkeit der Schlitze auf den entsprechenden Bussen erfolgt.

19. Die Vorrichtung gemäß Anspruch 14 oder 18, die ferner Mittel zur Initialisierung der von den Verzögerungsmitteln (71, 87) vorgenommenen Verzögerung als Reaktion auf zwei verschiedene Initialisierungsmarken (M1, M2) aufweist, die von der Kopfstation in einem bestimmten zeitlichen Abstand voneinander übertragen werden, und die Initialisierungsmittel folgendes aufweisen:

- Mittel (111, 175, 181, 183, 185, 187), um auf einem (B) der Busse eine erste der beiden Marken (M1, M1') zu erkennen, und um ein erstes Zeitsignal (127) auszulösen, nachdem die erste Marke erkannt worden ist, um Anforderungszählwerte in die Verzögerungsmittel (71, 87) zu schreiben, und
- Mittel (129, 154, 157, 159, 161, 163), um auf dem anderen (A) Bus die zweite Marke (M2) zu erkennen, und um ein zweites Zeitsignal (147) auszulösen, nachdem die zweite Marke erkannt worden ist, um Anforderungszählwerte aus den Verzögerungsmitteln zu lesen, um sie in die Anforderungswarteschlangen-Speichermittel (41, 85) einzufügen.

20. Die Vorrichtung gemäß Anspruch 14, die ferner Mittel (89, 90) aufweist, um sequentielle Zählwerte externer Anforderungen einerseits sowie sequentielle Zählwerte lokaler Anforderungen andererseits durch Addition zu summieren, die aus den Verzögerungsmitteln (71, 87) gelesen wurden, bevor sie in die Anforderungswarteschlangen-Speichermittel (41, 85) eingegeben werden.

21. Die Vorrichtung gemäß Anspruch 14, die in einem System eingesetzt werden kann, in dem mehrere Prioritätskategorien für Zugriffsanforderungen vorgesehen sind, und die Vorrichtung so viele parallele Anforderungswarteschlangen-Speichermittel und zugehörige Schaltungen aufweist, wie das System Prioritätskategorien hat, so daß Zugriffsanforderungen für jede einzelne Prioritätskategorie getrennt gespeichert und ausgeführt werden können.

Revendications

1. Procédé pour réguler l'accès à un système de transmission de données composé de deux bus de transmission à contre-courant et d'une pluralité de stations connectées entre elles, des tranches de temps séquentielles étant générées sur un premier des dits bus pour la transmission de données, chaque station transmettant des demandes d'accès pour les tranches de temps sur le second des dit bus, pour déterminer le droit de ladite station à transmettre des données dans une tranche de temps sur ledit premier bus, sur la base des demandes d'accès externes qu'elle a détectées en provenance des autres stations et de ses propres demandes d'accès locales;
ledit procédé comprenant l'étape consistant à :

- permettre, dans les dites demandes d'accès transmises sur le second bus, la réservation de tranches multiples par une station,

et les étapes suivantes à exécuter dans chaque station :

- gérer une file d'attente de demandes d'accès (41) des dites demandes d'accès externes et locales reflétant leur relation temporelle;
- introduire une demande d'accès dans ladite file d'attente de demandes d'accès lors de sa transmission sur le second bus;
- fournir une temporisation spécifique à chaque station (71) pour les demandes d'accès à introduire dans ladite file d'attente de demandes d'accès, pour compenser les différences de temps de propagation existant pour les tranches de temps et les demandes d'accès selon les différentes stations du système.

2. Procédé selon la revendication 1 comprenant, de plus, les étapes consistant à :

- enregistrer séquentiellement les dites demandes d'accès, sous forme de nombres représentant chacun un compte de demandes externes ou un compte de demandes locales, dans une pile FIFO de demandes (41);

- transférer le contenu de la position supérieure de ladite pile FIFO de demandes dans un registre de décrémentation (93);
- décrémenter le contenu du dit registre de décrémentation pour chaque tranche vide détectée sur ledit premier bus par la station; et
- lorsque le contenu du dit registre de décrémentation a atteint la valeur zéro, transférer le contenu de la position suivante de la pile FIFO de demandes dans ledit registre de décrémentation.

3. Procédé selon la revendication 1 comprenant, de plus, l'étape consistant à :
- fournir ladite temporisation spécifique à chaque station en enregistrant les demandes d'accès, avant de les introduire dans ladite file d'attente de demandes d'accès (41), dans une pile FIFO à retard (71) contenant un nombre prédéterminé d'entrées déterminant la quantité de temporisation.

4. Procédé selon la revendication 3 comprenant, de plus, les étapes consistant à :
- initialiser la temporisation effectuée par ladite pile FIFO à retard dans chaque station, en transmettant sur ledit premier bus deux repères d'initialisation (M1, M2) séparés par un intervalle de temps présélectionné (b);
- lancer dans chaque station, lors de la détection du premier repère sur le second bus, un premier signal de cadencement (127) dérivé des tranches du dit second bus, pour contrôler l'écriture des demandes d'accès dans ladite pile FIFO à retard, et, introduire une entrée dans ladite pile FIFO à retard pour chaque tranche de temps passant sur ledit second bus, cette entrée représentant un compte des demandes contenues dans une tranche de temps ou étant égale à zéro si la tranche de temps ne contient pas de demande d'accès; et
- lancer dans toute station, lors de la détection du second repère sur le premier bus, un second signal de cadencement (145) dérivé des tranches sur ledit premier bus, pour contrôle la lecture d'une entrée de ladite pile FIFO à retard, par tranche de temps, sur ledit premier bus, en vue de son transfert dans ladite file d'attente de demandes d'accès.

5. Procédé selon la revendication 4, dans un système qui comprend deux bus distincts, et dans lequel les tranches de temps se propagent uniquement le long du bus sur lequel elles sont générées, comprenant l'étape consistant à :
- transférer le premier repère d'initialisation (M1), lorsqu'il a atteint la fin du bus sur lequel il a été généré, dans sa forme originale ou modifiée, sur l'autre bus, en sorte qu'il puisse également se propager le long de l'autre bus.

6. Procédé selon la revendication 3 comprenant, de plus, l'étape consistant à :
- accumuler (89, 90) les demandes d'accès externes consécutives qui sont lues de ladite pile FIFO à retard, et les entrer sous la forme d'un compte unique de demandes d'accès dans ladite file d'attente de demandes d'accès.

7. Procédé selon la revendication 1, dans un système fournissant une pluralité de priorités pour demander l'accès aux tranches de temps, comprenant l'étape consistant à :
- générer chacune des dites tranches de temps avec une indication de priorité;
- insérer une demande d'accès pour une priorité donnée uniquement dans une tranche de temps portant l'indication de priorité respective; et
- fournir une file d'attente de demandes d'accès distincte pour chacune des dites priorités, pour recevoir les demandes d'accès externes et les demandes d'accès locales de la priorité respective.

8. Procédé selon la revendication 1 comprenant, de plus, les étapes consistant à :
- transmettre une demande d'accès locale en l'insérant dans une tranche de temps sur le second des dits bus, et
- copier chaque demande d'accès détectée dans une tranche passante sur ledit second bus, y-compris toute demande d'accès locale qui vient d'être transmise par la station respective, en vue de son insertion dans la file d'attente de demandes d'accès de la station respective.

9. Procédé pour réguler l'accès à un réseau de transmission de données, comprenant deux bus de transmission unidirectionnelle à contre-courant, une pluralité de stations connectées, chacune, aux deux bus, et au moins une station de tête de réseau pour générer des tranches de temps se propageant sur

les dits bus, chaque tranche étant composée d'un champ de contrôle d'accès (ACF) et d'un champ de segment d'informations, caractérisé en ce que chaque station demande l'accès à une tranche de temps en insérant une demande d'accès dans un sous-champ de demande du dit champ de contrôle d'accès d'une tranche passante, et détermine son droit à accéder à une tranche de temps pour la transmission de données sur la base de ses demandes d'accès locales et des demandes d'accès externes provenant d'autres stations qu'elle a détectées dans les tranches passantes,
ledit procédé comprenant l'étape consistant à :

- permettre, dans ledit sous-champ de demande, la réservation de tranches multiples;

et comprenant les étapes suivantes à exécuter dans chaque station :

- tenir un registre des demandes d'accès locales (LOC-REQ) générées par la station respective et des demandes d'accès externes (EXT-REQ) détectées en provenance des autres stations, sous la forme d'une file d'attente FIFO de demandes (41), ledit registre reflétant l'ordre séquentiel d'occurence des dites demandes d'accès;
- introduire un compte de demandes pour chaque demande d'accès locale à une ou plusieurs tranches de temps dans un sous-champ de demande vide d'une tranche passante;
- introduire, ensuite, un compte des demandes locales respectives dans la file d'attente FIFO de demandes; et
- contrôler le contenu du sous-champ de demande de chaque tranche passante, et copier toute demande d'accès externe et l'acheminer (91), sous forme de compte de demandes externes, en vue de son introduction dans la file d'attente FIFO de demandes.

10. Procédé selon la revendication 9 comprenant, de plus, les étapes consistant à :

- décrémenter le compte de demandes d'accès dans la position supérieure de la file d'attente de demandes de chaque station, lors de la détection d'une tranche libre en entrée, pour retirer une demande de tranche spécifique; et
- introduire, initialement, une temporisation spécifique à chaque station (71) sur chaque file d'attente FIFO de demandes, en sorte que, lorsqu'une demande de tranche spécifique est sur le point d'être retirée de la file d'attente de demandes d'une station, la station concernée détecte le passage, à son entrée, de la même tranche spécifique qui était ou sera détectée par toute autre station, sur son entrée, lorsque la même demande de tranche spécifique est sur la point d'être retirée de la file d'attente de demandes de ladite autre station.

11. Procédé selon la revendication 10 comprenant, de plus, l'étape consistant à :

- fournir ladite temporisation spécifique à chaque station, en enregistrant les demandes d'accès, avant des les introduire dans ladite file d'attente FIFO de demandes, dans une pile FIFO à retard (71) contenant un nombre prédéterminé d'entrées déterminant la quantité de temporisation.

12. Procédé selon la revendication 11 comprenant, de plus, les étapes consistant à :

- initialiser la temporisation effectuée par ladite pile FIFO à retard dans chaque station, en transmettant sur ledit premier bus deux repères d'initialisation (M1, M2) séparés par un intervalle de temps présélectionné (b);
- lancer dans chaque station, lors de la détection du premier repère sur le second bus, un premier signal de cadencement (127) dérivé des tranches du dit second bus, pour contrôler l'écriture des comptes de demandes dans ladite pile FIFO à retard, et, introduire une entrée dans ladite pile FIFO à retard pour chaque tranche de temps passant sur ledit second bus, cette entrée représentant un compte des demandes contenues dans une tranche de temps ou étant égale à zéro si la tranche de temps ne contient pas de demande d'accès; et
- lancer dans toute station, lors de la détection du second repère sur le premier bus, un second signal de cadencement (145) dérivé des tranches sur ledit premier bus, pour contrôle la lecture d'une entrée de ladite pile FIFO à retard, par tranche de temps, sur ledit premier bus, en vue de son transfert dans ladite file d'attente FIFO de demandes.

13. Procédé selon la revendication 10 comprenant, de plus, l'étape consistant à :

- accumuler (89, 90) les demandes d'accès externes qui sont lues consécutivement de ladite pile FIFO à retard sans intervention de demande d'accès locale, avant de les entrer dans ladite file d'attente FIFO de demandes, et accumuler, également, les demandes d'accès locales qui sont lues de ladite pile FIFO à retard sans intervention de demande d'accès externe, avant des les entrer dans ladite file d'attente FIFO de demandes.

14. Réseau de transmission de données comprenant deux bus de transmission unidirectionnelle à contre-courant, une pluralité de stations connectées aux deux bus, et au moins une station de tête de réseau pour générer des tranches de temps sur l'un des dits bus, dans lequel chaque station demande l'accès à une tranche de temps en insérant une demande d'accès dans une tranche passante; tient un registre des demandes d'accès en attente; et détermine son droit d'accéder à une tranche libre pour la transmission de données, sur la base de ses propres demandes d'accès locales et des demandes d'accès externes qu'elle a détectées en provenance des autres stations dans les tranches passantes, chacune des dites demandes d'accès comprenant une indication du nombre de tranches demandées. ledit réseau comprenant un dispositif d'accès multiple réparti à file d'attente dans chaque station, qui comprend :

- un dispositif de mémorisation de demandes en file d'attente (41, 85) pour enregistrer en ordre séquentiel les valeurs de compte de demandes représentant, respectivement, les demandes d'accès externes (EXT-REQ) et les demandes d'accès locales (LOC-REQ).
- un dispositif de temporisation (71, 87) connecté en série avec ledit dispositif de mémorisation de demandes en file d'attente pour fournir, pour chaque station respective, une temporisation sélective pour les valeurs de compte de demandes introduites dans ledit dispositif de mémorisation de demandes en file d'attente, pour compenser les différences de retard de propagation pour les tranches qui existent entre les différentes stations.
- des dispositifs d'insertion (103, 113, 115; 91, 117, 119) pour introduire une valeur de compte de demandes locales dans une tranche passante sur l'un des dits bus, et pour introduire la même valeur de compte de demandes locales dans ledit dispositif de temporisation (71, 87) sur ledit dispositif de mémorisation de demandes en file d'attente (41, 85) lors de l'introduction de ladite même valeur de compte de demandes locales dans une tranche passante sur ledit bus.

15. Dispositif selon la revendication 14, dans lequel ledit dispositif de mémorisation de demandes en file d'attente (41, 85) est une pile FIFO de laquelle la valeur de compte de demandes la moins récemment enregistrée est toujours extraite en premier.

16. Dispositif selon la revendication 14 comprenant, de plus, un dispositif de décrémentation (93) pour recevoir une valeur de compte de demandes extraite du dit dispositif de mémorisation de demandes en file d'attente (41, 85) et pour décrémenter son contenu d'une unité jusqu'à l'occurence d'un signal de commande respectif (131) transmis par un contrôleur de bus (81) qui détecte l'occurence de tranches vides, et pour fournir un autre signal de commande (151) au dit dispositif de mémorisation de demandes en file d'attente lorsque le contenu du dit dispositif de décrémentation est devenu zéro, afin de lancer le transfert d'une autre valeur de compte de demandes.

17. Dispositif selon la revendication 14, dans lequel ledit dispositif de temporisation (41, 87) est une pile FIFO à retard (41, 87) de laquelle la valeur de compte de demandes la moins récemment enregistrée est toujours extraite en premier.

18. Dispositif selon la revendication 17 comprenant, de plus, un dispositif (111, 175, 181, 183, 185, 187) pour dériver un premier signal de cadencement (172) pour contrôler l'écriture des valeurs de compte de demandes dans ladite pile FIFO à retard (71), à partir des tranches de l'un (B) des deux bus, et un dispositif (129, 154, 157, 159, 161, 163) pour dériver un second signal de cadencement (147) pour contrôler la lecture des valeurs de compte de demandes de ladite pile FIFO à retard, à partir des tranches de l'autre (A) des deux bus, en sorte que l'écriture dans et la lecture de ladite pile FIFO à retard aient lieu à la cadence des tranches de temps sur les bus respectifs.

19. Dispositif selon la revendication 14 ou 18 comprenant, de plus, un dispositif pour initialiser la temporisation du dit dispositif de temporisation (71, 87) en réponse à deux repères d'initialisation distincts (M1, M2) qui sont transmis par ladite station de tête de réseau avec un intervalle de temps donné entre eux; ledit dispositif d'initialisation comprenant :

- un dispositif (111, 175, 181, 183, 185, 187) pour détecter, sur l'un des dits bus (B), un premier des dits repères (M1, M1') et pour lancer, lors de la détection du dit premier repère, un premier signal (127) pour écrire des valeurs de compte de demandes dans ledit dispositif de temporisation (71, 87); et
- un dispositif (129, 154, 157, 159, 161, 163) pour détecter, sur l'autre des dits bus (A), le second des dits repères (M2) et pour lancer, lors de la détection du dit second repère, un second signal

(147) pour lire des valeurs de compte de demandes du dit dispositif de temporisation, en vue de leur insertion dans ledit dispositif de mémorisation de demandes en file d'attente (41, 85).

20. Dispositif selon la revendication **14** comprenant, de plus, un dispositif (89, 90) pour accumuler par addition les valeurs de compte de demandes externes séquentielles, d'un côté, et les valeurs de compte de demandes locales séquentielles, d'un autre côté, qui ont été lues du dit dispositif de temporisation (71, 87), avant de les introduire dans ledit dispositif de mémorisation de demandes en file d'attente (41, 85).

21. Dispositif selon la revendication 14, en vue de son utilisation dans un système comprenant de multiples catégories de priorité pour les demandes d'accès; le dispositif comprenant autant de dispositifs en parallèle de mémorisation de demandes en file d'attente et de circuits associés que le système possède de catégories de priorité, pour permettre un enregistrement et une gestion séparée des demandes d'accès pour chacune des catégories de priorité.

FIG. 1A DUAL BUS CONFIGURATION

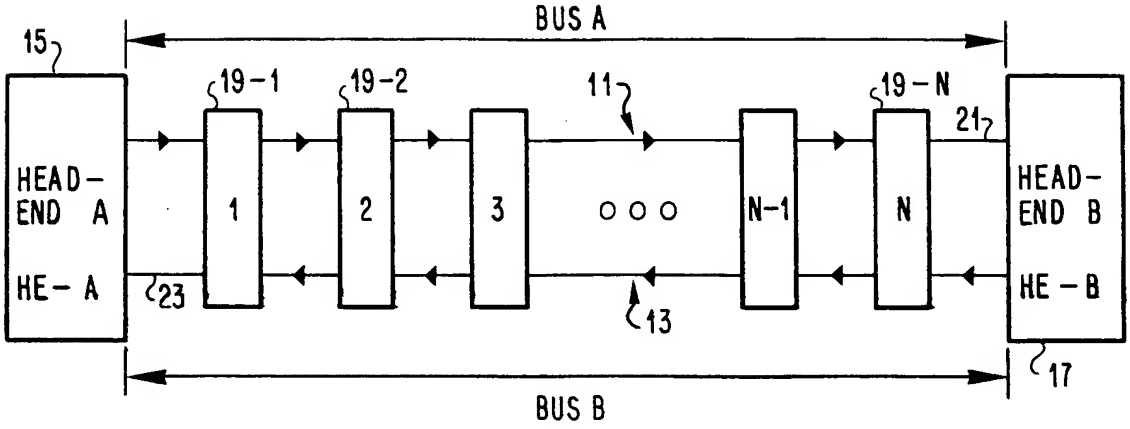


FIG. 1B FOLDED BUS CONFIGURATION

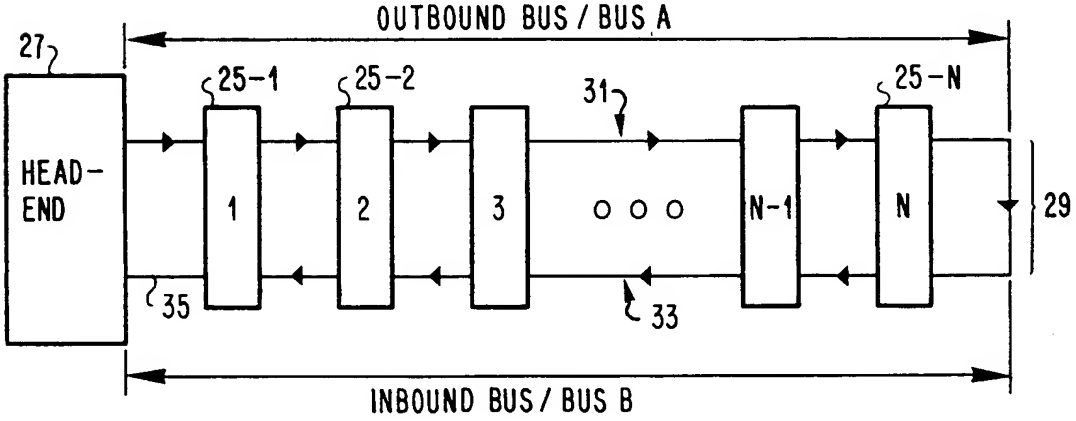


FIG. 2A CYCLE FRAME

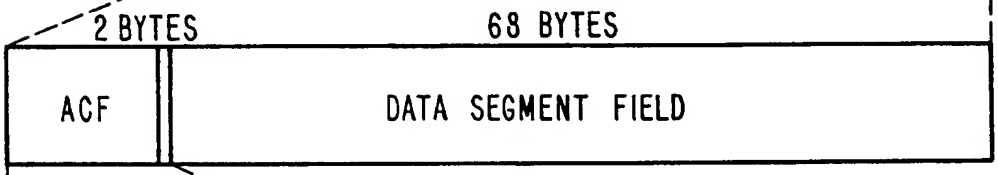


FIG. 2B SLOT FORMAT

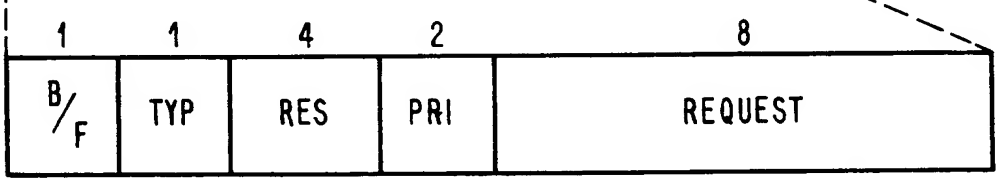


FIG. 2C ACCESS CONTROL FIELD ACF

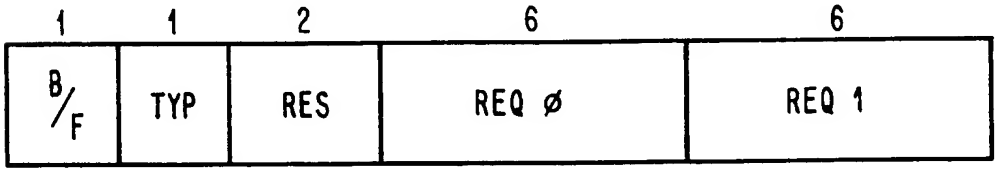
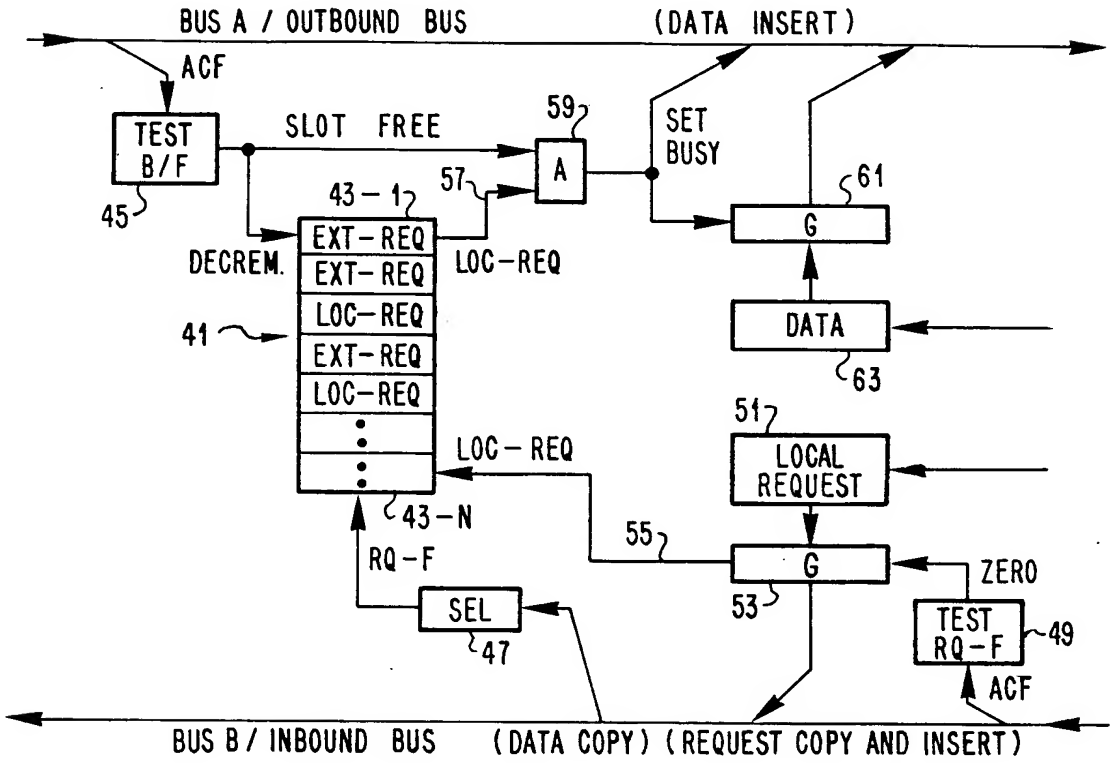


FIG. 2D ACF WITH MULTIPLE REQUEST FIELDS

FIG. 3 BASIC FIFO REQUEST QUEUE



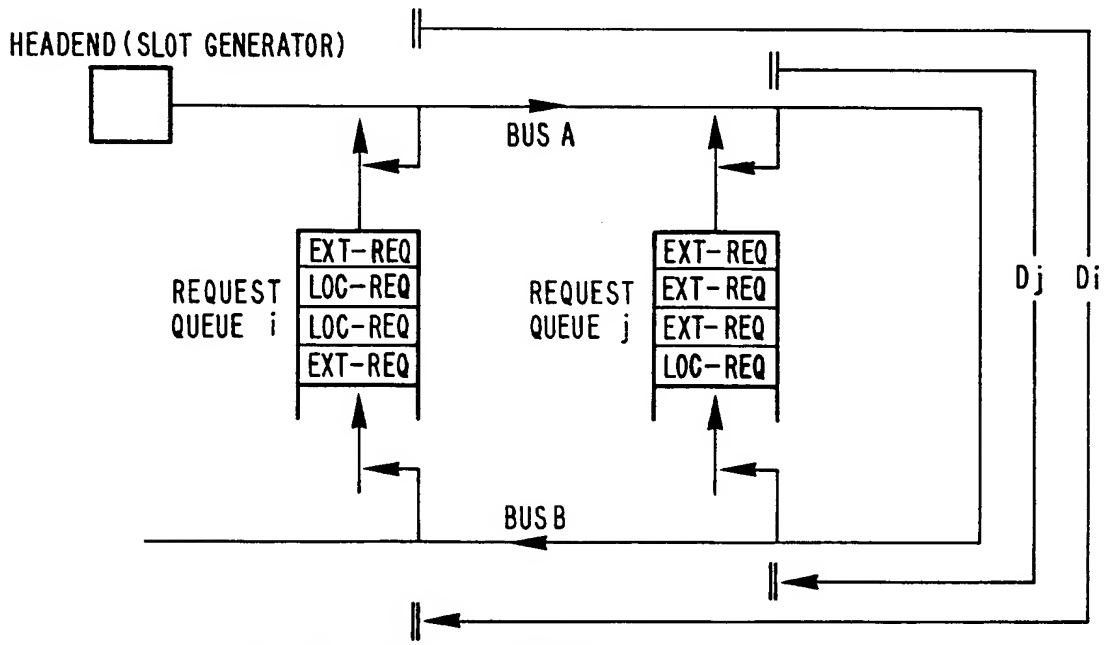


FIG. 4 PLURAL STATION FIFO QUEUES

FIG. 5 REQUEST QUEUE WITH DELAY

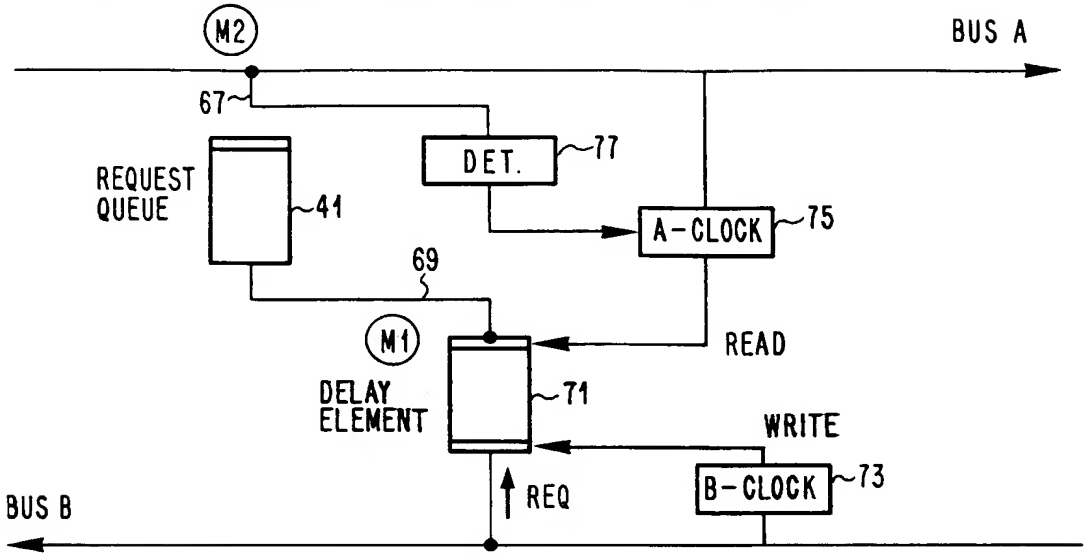


FIG. 6 PLURAL STATION FIFO QUEUES WITH DELAY

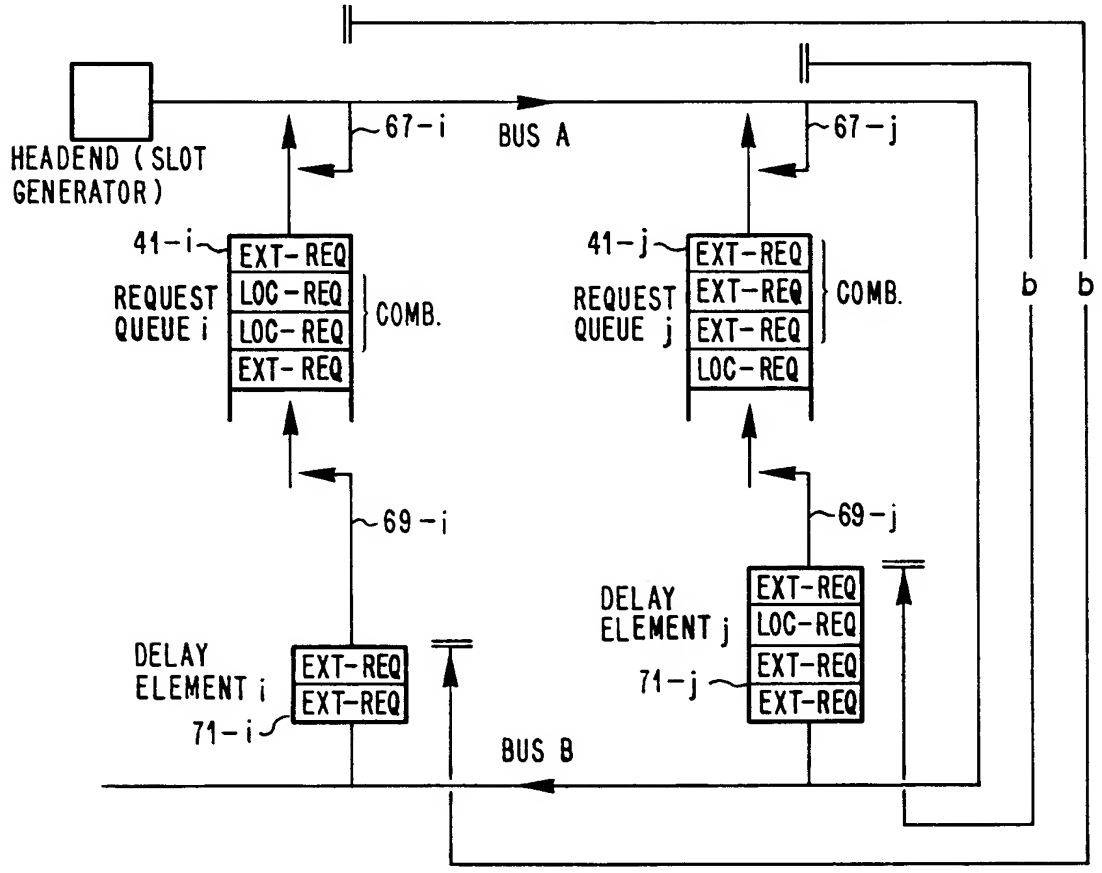


FIG. 7 A

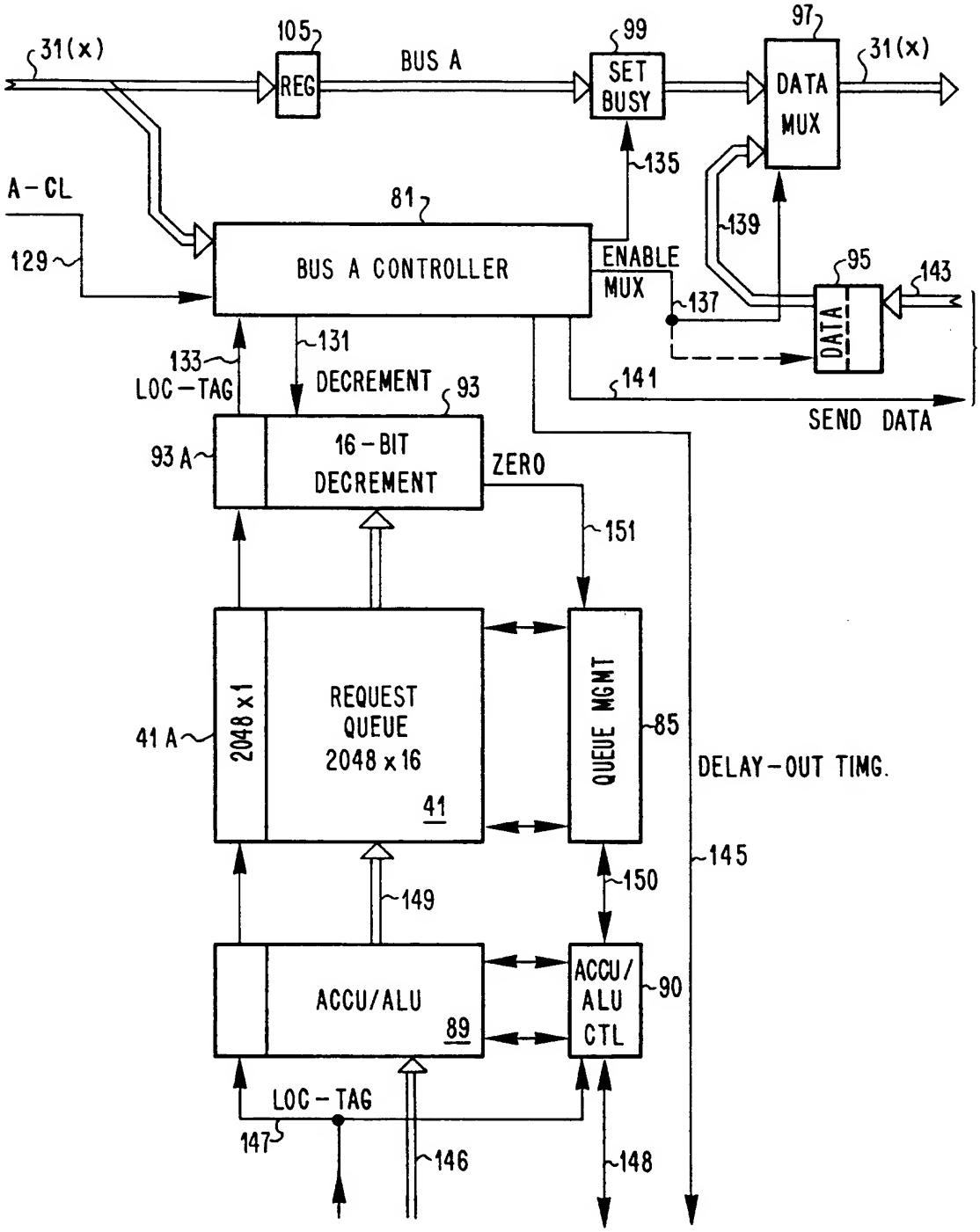

31(x)
105
REG
BUS A
99
SET BUSY
97
DATA MUX
31(x)
135
A-CL
81
129
BUS A CONTROLLER
ENABLE MUX
139
95
143
137
DATA
131
133
LOC-TAG
DECREMENT
93
141
SEND DATA
93 A
16-BIT DECREMENT
ZERO
151
41 A
2048 x1
REQUEST QUEUE 2048 x 16
41
QUEUE MGMT
85
DELAY-OUT TIMG.
145
149
150
ACCU/ALU
89
ACCU/ ALU CTL
90
LOC-TAG
147
146
148

FIG. 7B

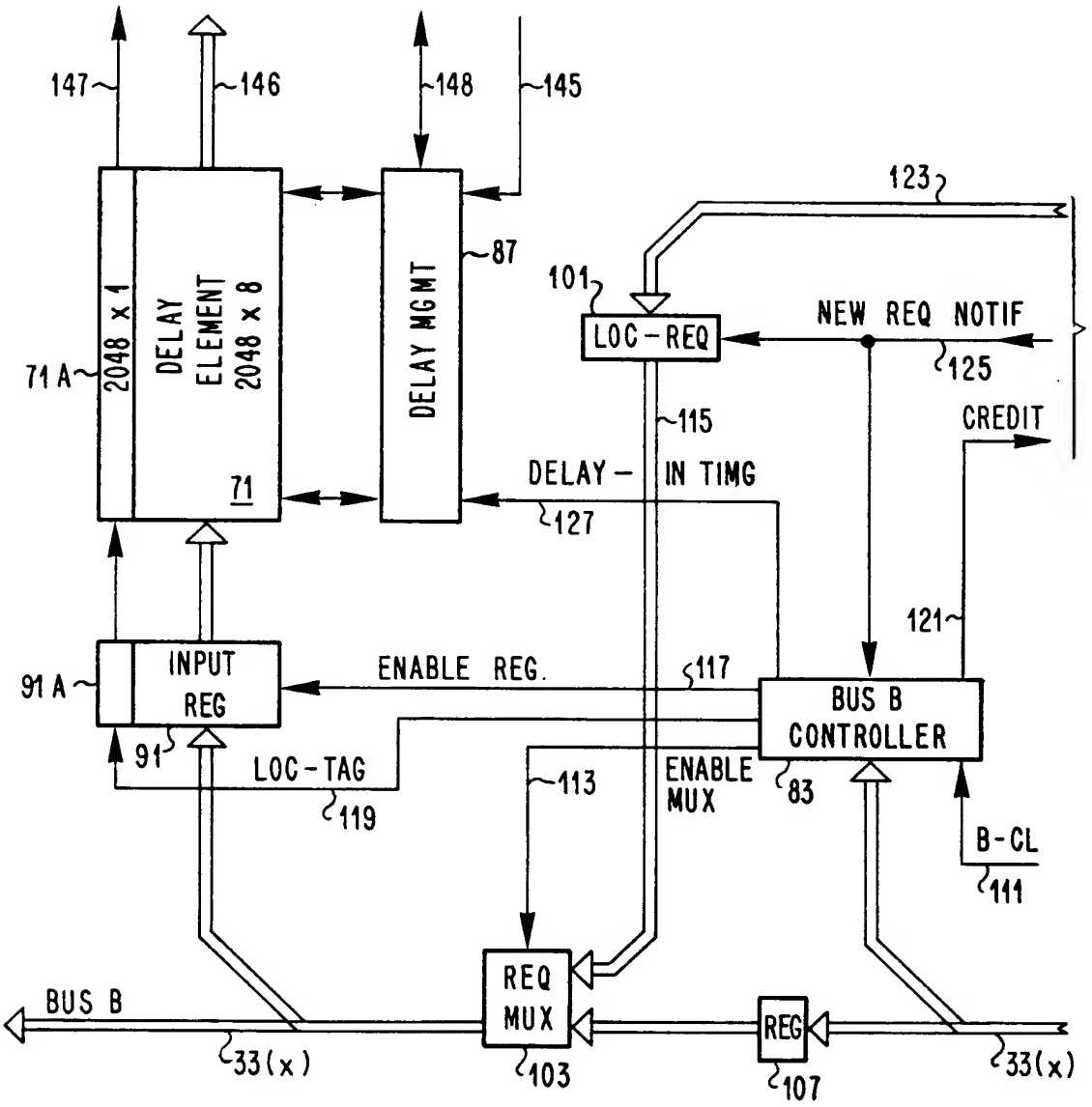

147
146
148
145
123
87
101
DELAY MGMT
LOC-REQ
NEW REQ NOTIF
125
71A
2048 x 1
DELAY ELEMENT 2048 x 8
71
115
CREDIT
DELAY-IN TIMG
127
121
91A
INPUT REG
ENABLE REG.
117
BUS B CONTROLLER
91
LOC-TAG
119
113
ENABLE MUX
83
B-CL
111
REQ MUX
BUS B
33(x)
103
REG
107
33(x)

FIG. 8 BUS A CONTROLLER

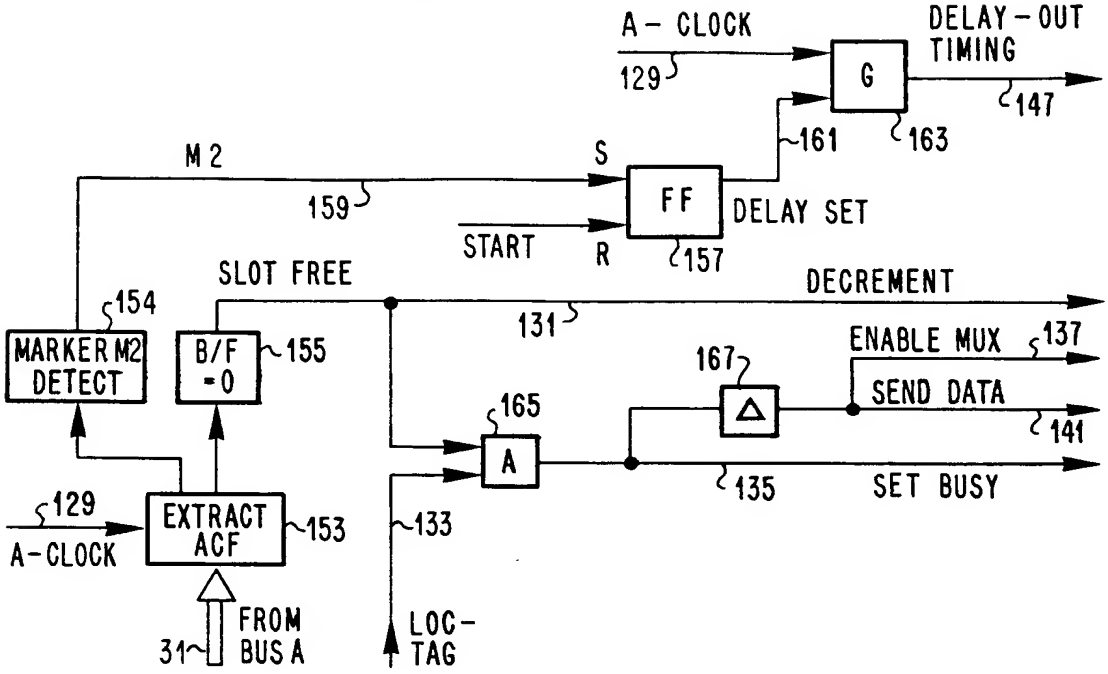
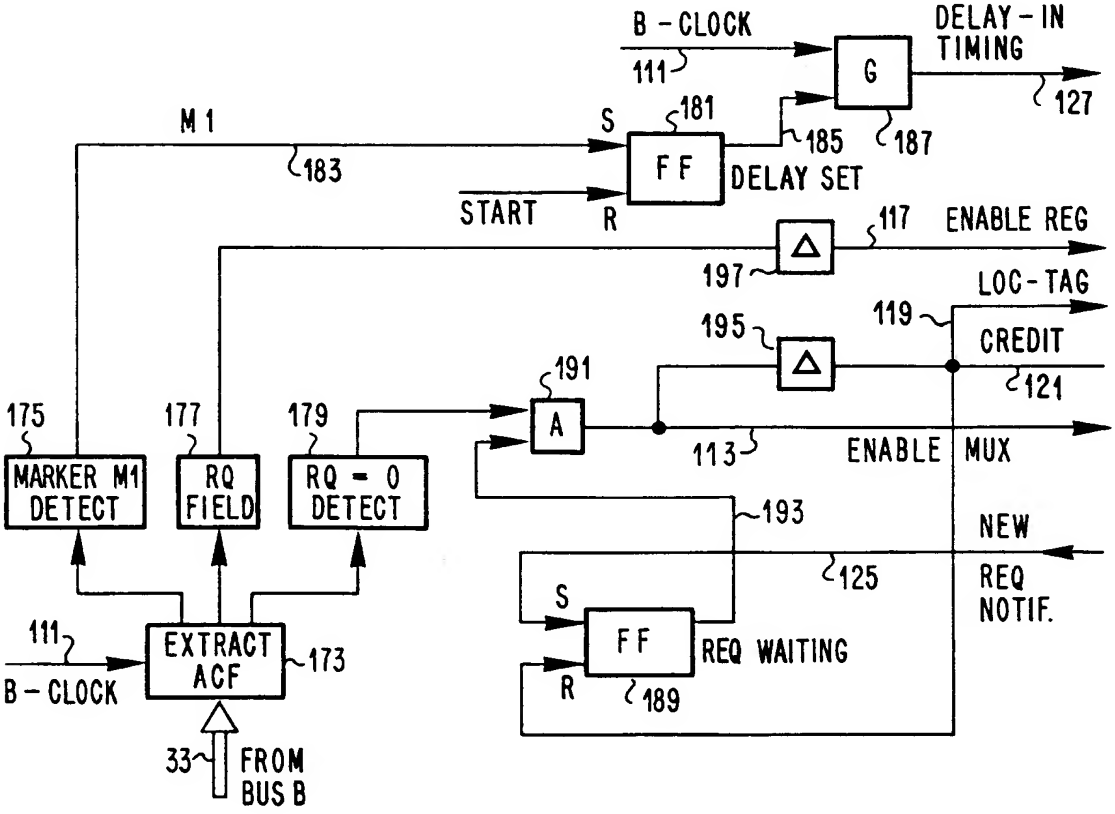


FIG. 9 BUS B CONTROLLER



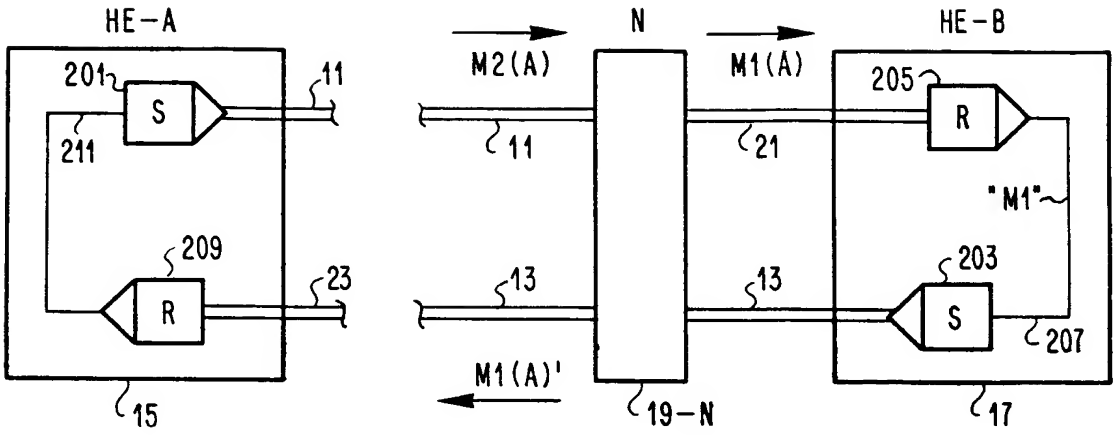


FIG. 10 WRAP-AROUND IN DUAL BUS SYSTEM

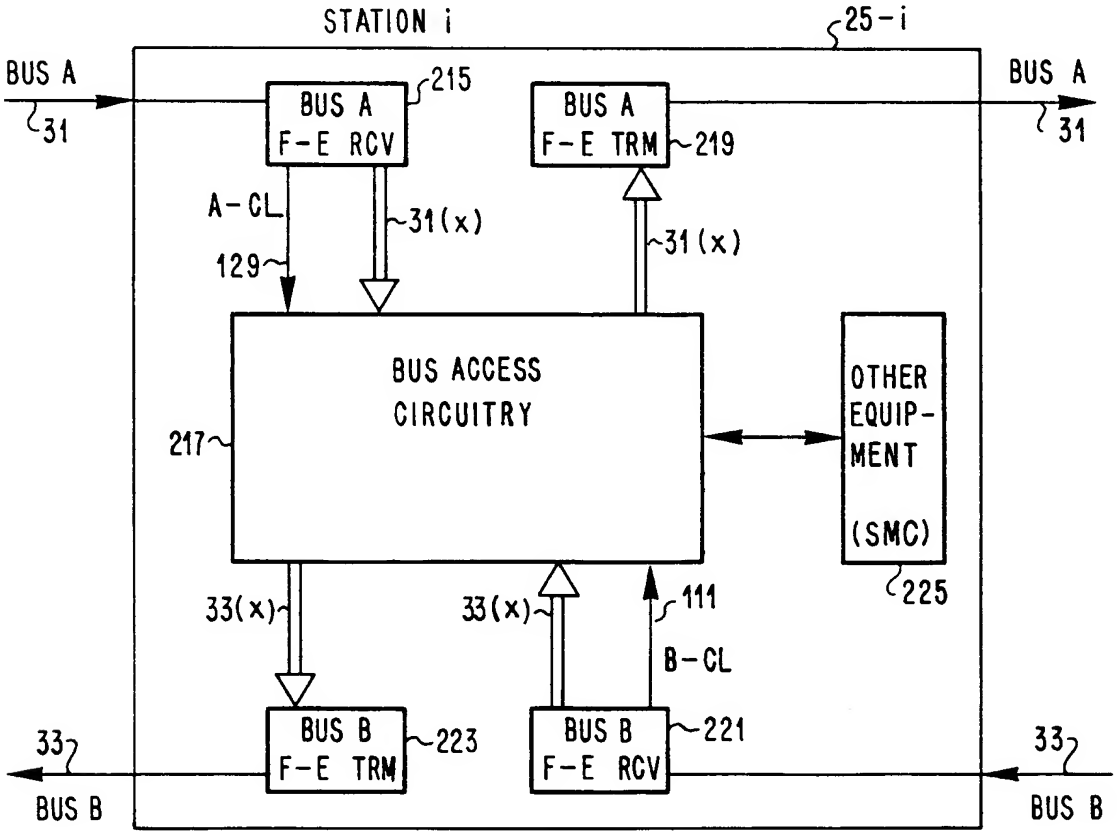


FIG. 11 SIGNAL CONVERSION BETWEEN BUS AND STATION (NODE)